ETHERISC

White Paper

1.11.2017

Table of Contents

Version Control

Version	Date	Author	Remarks
0.5	15.04.2017	Christoph Mussenbrock	Initial (from old WP 0.3)
0.6	04.10.2017	Christoph Mussenbrock	Rework structure, integrate Token mechanics.
0.7	16.10.2017	Christoph Mussenbrock	Rewrite, compact, extend
1.0	01.11.2017	Christoph Mussenbrock	Ready for publishing

Acknowledgements

The ideas in this paper are a collaborative result of many talks and discussions, online and in person, with some brilliant minds of the crypto-economic space. The Etherisc team would like to say “thank you” especially to the following people, who added considerable value to the draft: Ron Bernstein, Jake Brukhman, Alexander Bulkin, Alex Felix, William Mougayar, Micah Zoltu and last but not least all those members of our gitter channel, who gave valuable feedback and criticism and encouraged us to follow our path.

1 About this document

As a result of the Nov/Dec 2016 Hackathon, the team wrote a White Paper, which was released to public in its Version 0.3. Focus of the hackathon was the attempt to outline the core of a blockchain based reinsurance market.

In the meantime, some things have changed and clarified.

First, we have seen that the development of an open protocol will be at the core of our efforts. This leads to a much broader approach, needs different legal entities and more funding. We have consequently increased our funding goal.

Second, we have seen that a reinsurance market will most probably not be the first step. Trading of risks (and tokenization of risks as a precondition) is a different regulatory area and probably more difficult to achieve; furthermore, without first establishing a regular insurance business we don't have suitable risks to trade.

Third, we have seen that we need to apply for an insurance license in some relevant jurisdictions, namely in the DACH area (Germany, Austria, Switzerland) or in some offshore countries like Malta or Bermudas, and additionally we try to establish an insurance business in the US on a slightly different, but still legal base.

These three cornerstones don't reflect in the former White Paper, which was centered around the tokenization of risks.

This White Paper is the attempt to structure the overall picture of decentralized insurance along the new insights.

2 Why is insurance a candidate for decentralization?

The multi-trillion dollar insurance industry is dominated by huge corporations, weighed down by heavy regulation and plagued by misalignments of company and consumer incentives. The insurance world has devolved into an inefficient, expensive and ultimately frustrating industry. When customers most need help, they can end up fighting in vain for reimbursement from companies whose profits too often depend on avoiding paying out.

Etherisc is building a platform for decentralized insurance applications. With visionaries like you, we can create a platform full of opportunities across the industry's value chain. Corporates, large and small, not-for-profit groups and insurtech startups can all come together to provide better products and services. We aim to use blockchain technology to help make the purchase and sale of insurance more efficient, enable lower operational costs, provide greater transparency into the industry and democratize access to reinsurance.

Blockchain can provide the means to disintermediate the market with a peer-to-peer risk platform that helps insurance return to its roots as society's safety net. We even envisage new groups building their own bespoke insurance risk pools and services on the platform. And Etherisc will be a fully-compliant, fully licensed insurance platform for the emerging blockchain economy.

In short, Etherisc can deliver the insurance industry the modernization customers are crying out for.

We have assembled an award-winning team of experts, experienced in delivering innovative products. We have already demonstrated the use-case for decentralized insurance applications with a successful flight-delay DApp that debuted at one of blockchain's biggest international conferences. This was the first insurance product live on a public blockchain. With your support, we can now build out our open-access platform and help make one of the globe's biggest industries finally work the way it should – for everyone, everywhere.

3 Analysis of Basic Insurance Paradigms

3.1 Overview

Since our early prototype in September 2016, we analysed the basic principles of insurance and developed a token system on top of these principles, which is sustainable and sound (later in this paper we will describe which kind of tokens we consider to be "sustainable" and "sound").

First, we analyze insurance and and break costs and capital flows down into three elements:

1. Expected value of risk
2. Capital costs for long tail risks
3. Transaction costs

We show that the first isn't a source of profit, because it is only a redistribution of capital corresponding to sharing risks among the participants.

The second are a source of fixed income, at a certain risk. Capital has to be locked for a certain period in time, and there is a potential risk of losing the capital provided, e.g. in the case of a rare but catastrophic event, also known as "black swan event". Capital providers are compensated for this risk. This compensation is calculated based on the lock-up time and on the risk of what is being insured.

The third are a source of entrepreneurial revenue and increase with higher efficiency of the business processes.

We argue that today insurance companies are the predominant way to organize these elements and that blockchain technology provides an opportunity to replace insurance firms by decentralized structures using a standardized protocol. Capital and revenue streams can then be represented by tokens.

Our conclusion from this analysis is that we need two types of tokens. The first one supports the coordination and economical incentivization of actors in a decentralized insurance system. This is the token to be discussed for a token sale to fund the development of a protocol and platform for decentralized insurance. We call these "protocol tokens".
The second type of token represents risks - this type will come as a collection of similar tokens, one for each risk pool, we call those "risk pool tokens". These "risk pool tokens"

will be discussed in a separate document, as they underlie a different economic dynamic.[1]

In a distributed environment with many participants, building products as a collaborative effort, the protocol token serves as glue, as collateral, and as representation of the material and immaterial value of the network, much as Ether serves as a means to secure the stability of the Ethereum Blockchain.

In Chapter 3, we detail the DIP protocol token. Chapter 4 shows a concrete example of the use of the token in an insurance context.

3.2 Principles of insurance

Lots of literature has been written on the theory of insurance, but the basic principles are simple. Let's start with an example . The example is of course simplified, and serves the sole purpose to explain the principle.

We consider a homeowners insurance. Insurance is about probabilities of losses, so it would be interesting to see what the probability of a damage is. A homeowners insurance typically covers a number of perils, including fire, natural disasters, water, and even falling objects[2]. But it is difficult to obtain real numbers, as insurance companies are not very transparent with their fundamental data[3]. We will assume that, for our example, the probability is 0.1%.

For our fictional example, let's assume insurance had not been invented yet. In this fictional world, Alice owns a house. The house is worth $100K. The probability of a complete disaster is 0.1% per year (that is one devastating event in 1,000 years). Alice wants to ensure that she has access to enough funds to get a new house in the case of a disaster. So she decides to get a loan of $100K and has to pay redemption (also called principal) and interest rate.
Additionally, she pays an interest rate of maybe 1%, so she has yearly costs of $1,100 ($100,000 loan * 1% interest rate plus $100 annual redemption = $1100.00).
Now we show how pooling risks in an insurance scheme reduces these costs drastically.

---

[1] In https://www.etherisc.com/whitepaper, we already described a possible implementation of a "risk pool token", which aggregate similar risks and which can be sold and traded to provide the necessary funds to cover "long-tail-risks".
[2] Allstate.com: What Perils Are Typically Covered By A Homeowners Insurance Policy?
[3] A quick market survey in Germany shows that you get a homeowners insurance for considerably less than 0.1% of the value. For simplicity, we'll assume that the premium is 0.1% plain and we don't take insurance taxes etc. into account.
From the relation premium/value, we can easily estimate an upper bound for the probability. One of the most fundamental principles of insurance is that the expected losses should not surpass the collected premiums ("Risk loading" - cf. http://www.wiley.com/legacy/wileychi/eoas/pdfs/TAP027-.pdf). The expected losses are - simplified - number of policies multiplied with the probability of loss multiplied with the loss (which is equal to the value), and collected premiums are number of policies multiplied with premium per policy. It follows that the probability can be approximated by premium/value, which is lower than 0.1% in our market test.

3.2.1 Sharing the expected value of risk

Assume 100,000 homeowners are coming together in a pool. Again, everybody pays a $100 share; this amount is now called the “premium”. They collect a total of $10,000,000 in premiums. But now there is a difference to Alice, who takes care only for herself: because of the law of large numbers[4], with a very high probability there will only be about 100 fires, causing a damage of about $10,000,000! And because the sum of all premiums is also $10,000,000, the whole damage can be paid out of the collected premiums, there is no need for every house owner to take on a loan. (Because premiums are collected at the beginning of the year, and all the houses “expected” to burn don’t all burn at the beginning of the year, but more or less are equally distributed over the year(s), there is a so called “float”[5] of liquidity which can also generate a significant revenue. For simplicity, we won’t focus on this effect in this paper.

So the costs for each single house owner are now reduced from $1,100 to $100!
This difference asks for an economical explanation. Let’s have a closer look into it. First of all, if all house owners would follow Alice’s example, they would need a huge amount of loans, from which only a tiny part of 0.1% would been needed in the average. It is clear that providing unused liquidity is costly. Pooling of risks in an insurance optimizes the use of capital, and the participants benefit from the reduced costs, not to speak from the difficulties to obtain a loan without collateralization!
Second, if everybody only cares for himself, only a tiny fraction of participants are struck by disaster, and have the burden of actually paying back their loan. The others can pay back without loss, as soon as they don’t need protection. In an insurance collective, we have solidarity: with the premiums, everybody pays for the damages of the others.
To summarize, the risk pool offers three advantages for the participants:

1. Building a large liquidity pool.
2. Guaranteed access to this liquidity in case of a damage.
3. Mutual subsidizing of damages.

Such a pool may be designed solely to benefit its’ participants, and to not make any “profit”. If the pool did generate profits, these profits could be distributed back to the participants, effectively reducing the premiums again to a level where no profits are generated. Such an insurance would have a loss ratio of 100%, because all premiums are used to pay the losses.

This is the very basic effect of risk transfer in insurance. Please note that the effect increases with the pool size.

But still, this is not the whole story.

[4] https://en.wikipedia.org/wiki/Law_of_large_numbers
[5] http://www.npr.org/sections/money/2010/03/warren_buffett_explains_the_ge.html

3.2.2 Sharing the long-tail-risks

In some years, there are more fires, in other years, less. To account for these variations in damages, the whole pool has again to raise some money, e.g. $10M, to cover the unlikely event of a burst of many fires in one particular year. And let's suppose that the interest rate for this capital is even particularly high, e.g 20%. We will have total costs for this capital of $2M. The interest rate for the capital is a function of the risk and the riskless interest rate on the capital market; in an efficient market, the interest rate will compensate for the higher risk in comparison with a risk-free investment and will also contain a fair profit. So basically, this is where profits are generated for providing capital in an insurance structure.

The overall costs of $2M are distributed among all house owners, yielding an additional cost of $20 per house owner per year, which is added to the premium.

So after this, there is also a protection against "long tail risks" or "black swan events", at a cost of $20 per house owner. Again, the risk diversification effect increases with the pool size.

Overall, participants now pay $120 per year for their house insurance. The loss ratio is now reduced to 83% because of the capital costs of protecting the long tail risks[6].

3.2.3 Sharing the transaction costs

To organize 100,000 people in a pool, a professional structure is needed, otherwise, every single participant would have to talk to every other, which would simply be impossible. The operation of this professional structure adds transaction costs to the premium. This is the reason why insurance companies have come into existence: They provide a way to decrease transaction costs for the participants of the pool, creating an economy of scale and coordinating a huge number of participants and employees[7]. The effect is considerable and enables the modern form of insurance with huge customer bases and a capitalization which can cover even global catastrophic events like hurricanes and earthquakes. However, the remaining transaction cost are still considerable: a recent study by KPMG shows the impact on the loss ratio, which is about 66% in the average[8].

---

[6] $100 for covering the risk against $120 premium => 100/120 loss ratio = 83%

[7] The downside of this is the fact that inefficiencies tend to hide in the organization. The bigger the organization, the lesser people are doing the real work (people at the "rim" of the organization) and the more people are needed in the center to organize the people at the rim (the "management"). Furthermore, to limit internal inefficiencies, companies need a plethora of control mechanisms (that's the old style) or complicated incentive systems (that's the more modern way).

[8] https://assets.kpmg.com/content/dam/kpmg/au/pdf/2016/general-insurance-industry-review-2016.pdf

3.2.4 Information asymmetry

Together with the reduction of transaction costs comes an asymmetry of information, which leads to a further increase of costs and to incredible profits for the big insurance companies. The unbounded collection of customer data and the exclusive exploitation of this data is a consequence of this imbalanced relationship. It creates an “unfair competitive advantage” for existing companies: companies with big data vaults can offer better products, and thus further optimize their data base.
One of the core goals of a decentralized insurance platform is the disruption of this circle, giving back to customers the ownership of their data.

3.2.5 Summary

The three elements described above; pooling or risk, risk transfer, and efficient administration are necessary. You can’t have insurance without each of them.
For the purposes of this paper, I will call them:

1. expected value of the risk
2. capital costs for long tail risks
3. transaction costs

As we have seen, a community may not wish to generate profit from the first element. The second element yields a risk fee for binding capital which depends on the structure of the particular risk: It is typically lower if the risks are granular and uncorrelated; it is typically higher if the risks are clustered or correlated. The third one depends on the complexity of the process. A simple and highly standardized insurance “product” has a smaller transaction complexity than a more complicated, non-standardized product. This will reflect in lower transaction costs.

The three elements are completely independent of the underlying technology, economic environment or currencies. They are the atomic building blocks of every risk-sharing system[9].

As an additional aspect we have seen the information asymmetry which is inherent in the traditional insurance systems, and which is undesirable.

The distribution of expected value (element 1) and capital costs for long-tail-risks among participants (element 2) is inevitable and not specific for a blockchain solution. Therefore, let’s focus on the third element.

[9] There is a fourth element – reinsurance. The purpose of reinsurance is to reduce the cost of risk diversification by categorizing and securitizing different risks. Reinsurance and “wholesale” risk transfer enabled by reinsurance adds another layer of complexity, and therefore we won’t discuss reinsurance in this paper.

Blockchain is essentially - among other aspects - a way to solve the transaction cost problem without firms. Without the “design pattern”[10] of firms, transaction costs are subject to combinatorial explosion. The coordination costs for n participants are roughly of Order $O(n^2)$ and firms reduce this to $O(n)$. Because of this huge gain in efficiency, firms have many ways to hide profits in the transaction costs, and on the other side internal inefficiencies don’t show up fast.

Transaction costs also appear in another context: regulations, which are deemed necessary to protect customers in a context with built in conflicts of interest. Regulations form a very effective “competitor” barrier to entry. While insurance companies often complain about the burdens of regulations, they actually don’t have much interest in reducing these burdens, as they discourage new competitors from entering the market.

3.3 Blockchain can help to solve issues of traditional insurance

While the current insurance business has evolved over centuries, and is optimized in many aspects, we have seen that it has severe shortcomings to the disadvantage of customers. We will outline some properties of an alternative system, which remedies these shortcomings.

First, an alternative system should of course offer the basic ingredients of any insurance system: covering expected losses, covering long tail risks, and covering of necessary transaction costs. Obviously, we need ways to capitalize such a system, and we need a system to reduce transaction costs to a minimum. Transaction costs cannot be eliminated completely. But open markets have proven to be a solution for these challenges, and therefore, we propose a market-based approach with two components:

- an open marketplace for capitalization of risks
- an open marketplace for insurance related services

This is where blockchain comes into play: a decentralized solution on blockchain can implement such open marketplaces in a way that is collusion resistant and has no single points of failure. We can watch the emergence of many such marketplaces for different domains, like computation, file storage, exchange of assets; and insurance is just another domain in this respect.

More specific, blockchain can help to solve four main problems which pile up costs in traditional insurance companies:

1. Coordination (“managerial”) costs.
2. Conflict of interest between customers and company.
3. Information asymmetry between customers and company.
4. Access to risk pools

Advantage 1. In traditional firms, you have two types of employees: the first group is doing the actual work, the second group is coordinating the whole system. The larger a

[10] On the importance of design patterns, see also: Design Patterns, by the “Gang of Four” https://www.pearson.com/us/higher-education/program/Gamma-Design-Patterns-Elements-of-Reusable-Object-Oriented-Software/PGM14333.html#

company grows, the more energy flows in the second group (like a circle, the first group forms the rim of the circle, the second the area; the larger the circle, the less efficient are the processes, and the more energy flows into the coordination of the coordinators). Blockchain can help reduce these coordination costs. Instead of a posse of managers, “smart contracts”[11] can act as trustless hubs between the agents at the rim of the system, and thus eliminating most of the costs and the inefficiency of the management.

Advantage 2. In a traditional insurance company, the company “owns” the whole process, including the tasks which tend to raise conflicts of interest between customer and company. An obvious example is claims management: The claims manager has the explicit goal of minimizing payouts for damages, because he is employee of the insurance provider! Of course there is a guild of “independent” appraisers and experts, but who pays their bills?

Blockchain can solve this conflict of interest, by enabling truly independent experts (who for example may be publicly ranked by their reputation for efficiency or fairness), and whose work is independent of the insurance provider, as well as being transparent and auditable by the whole community.

The same is valid for another area, where the conflict of interest is (intentionally) not obvious; consider Product Design. An insurance company has a big advantage over customers, because they can design products in a way which perhaps unfairly maximizes revenues (sales) and minimizes payouts (expenses).

For example if a customer expects a payout from an insurance policy they bought for a particular “event” but the insurance company does not provide the payout because the company maintains that the policy bought doesn't actually cover that “event”, the customer experience is severely degraded and trust is eroded between consumers and insurance providers.

Advantage 3. Information asymmetry is in itself a source of inefficiency and high transaction costs. Insurance companies gather data and information in huge private silos by proprietary means and the data is often not shared., This data and the companies’ experience in analyzing the data is considered one of the main differentiators in the market. The reasoning behind decisions made based on this data is opaque and difficult to challenge. In a blockchain environment, all fundamental data and the decisions based on the data can be transparent and objectively validated.

[11] Some blockchains like Ethereum (which we use) enables programs (called “smart contracts”) that are un-censorable, immutable, and permanent. These smart contracts can interact with each other to perform a wide variety of actions, including financial and escrow transactions. This makes possible direct and transparent interactions between two parties who may be and may remain anonymous, that previously required a third-party intermediary to be effective. The term was originally coined by Nick Szabo, but in a slightly different meaning. Note: The above definition was thankfully supplied by Ron Bernstein, who was not successful in finding the original author - please contact us if you are the author.

Advantage 4. The risk pools of traditional insurances are attractive investment vehicles, but currently, they are not open to the public, and the profits generated benefit only a small circle of investors. Blockchain can democratize the access to similar instruments, by tokenizing risks with "Risk Pool Tokens", see our 2016 whitepaper for details. We will consider issuing of such tokens at a later point of time.

3.4 Requirements and consequences of a decentralized implementation

To offer an alternative to traditional company-centric insurance systems, we can identify some requirements and consequences for implementing a decentralized insurance protocol.

3.4.1 General requirements for decentralized insurance

1. We need a protocol and not just an (decentralized) application. Insurance is way too complex to be covered by a single application, and needs some means to incentivize participants to use it. Fostering "Network Effects"[12] is one such mean and can lead to a sustainable and growing user base.
   While a single contract can handle a single product, this singularity will not generate the network effects which are desirable to form multiple large pools of similar risks needed to get the benefits of the "law of large numbers" working. Decentralized insurance will work only if the value chain is decomposed and there is a way different participants can cooperate on the process in an interoperable way. A standardized protocol defines this way. The (architecture of the) protocol is the sole "central" part in the model.

2. A decentralized insurance protocol can replace "the firm", by implementing a standardized set of rules for how stakeholders in the system interact with smart contracts and with each other using the protocol. By this, most of the coordination costs are replaced by autonomous and automated contracts and procedures and enforce efficiency by open market mechanisms. At the same time, a protocol does not impose a fixed set of code to the participants, but allows for flexible extension and interpretation of the basic rules.

3. The development and operation of a protocol needs funding. Even if we can drastically reduce the coordination costs, there are still the costs for the initiation of the system - e.g. acquisition of licenses, development of smart contracts, audits, as well as costs for agents at the "rim" of the system which we cannot eliminate completely. Therefore we need a way to collect these costs from the ultimate customers and distribute them amongst these agents.

[12] Network effect is described as the effect that one user of a good or service has on the value of that product to other people. The classical example is telephone: the more people use it, the more valuable the telephone is for all.

4. We also need a way to calculate and distribute the expected value of the risk and the capital costs for covering long tail risks amongst the customers.

3.4.2 Requirements for token

1. **Tokenization**
   may be the solution for these requirements - but only if the token is intrinsically required for the protocol to operate efficiently; i.e. “baked into” the protocol itself ***and*** usage of the protocol is only possible via tokens. If the token were not intrinsic to the use of the platform, then some new actor could replicate the protocol except without the token, and migrate users to the new protocol without the friction of a purely "rent seeking" token.

2. **Protocol tokens:**
   For the *distribution of the transaction costs* we need a different type of token. This token has to be designed in a way that incentivizes the use and the efficiency of the protocol: the revenue associated with this token or its price should increase with the efficiency and use of the underlying processes.
   In the next chapter, we describe a proposal for a token with these properties.

3. **Risk Pool tokens:**
   For the distribution of the *expected value*, and for the *distribution of the capital costs for covering long-tail-risks*, we need a type of token which generates a foreseeable profit. The profit solely depends on the underlying risk structure, the number of risks, their correlation, and so on. The value therefore depends only on the knowledge of the risk parameters (which can be incomplete) and mathematics.
   These tokens will e.g. yield a fixed revenue or generate an equivalent rise in price for their owners (which is equivalent). This type of token will be implemented in a second step.

4. Now that we have elaborated the necessary token types, we can backtest if these tokens are “necessary”
   Etherisc will build an economic space for decentralized insurance
   The space will have a broad set of participants, customers, service providers, risk carriers, etc., the goal is to incentivize these participants to cooperate and behave well, and in line with the interests of the whole space.
   This space is difficult to build. It comes at a cost.

   What adds value to the space:

Building Block	Consists of	Resistance against forks & copycats
Licenses	Formal approval by authorities	Cannot be copied
Operational Model	Infrastructure to run a business	Cannot be copied
Products	Code (Frontend/backend) infrastructure Certifications/Audits Developers Product managers	Tech can be copied, but products are micro-ecosystems with development roadmap, user base, customer support, core development team, supporters and contributors
Users	Customers Supporters Contributors	Cannot be copied
Network	Formal or informal Relations to other projects,	Cannot be copied. Relations to other projects are based on common vision.

5. **Conclusion:**
   only tech can be copied easily. Most of the value-bearing components of the economical space (the value that participants bring) can't be copied easily
   The economical space will offer opportunities to generate profits. These profits should benefit those who have participated to build up the space, and they will expect the platform to protect their participation.
   Reason:
   If you have two identical platforms, one platform with some kind of protection mechanism for its creators and contributors and one platform without such protection. The platform with protection will of course attract more contributors. It will have the stronger network effects.
   A platform without protection is subject to the "Tragedy of the Commons"
   In the prospect of decentralized exchanges the use of a token is no longer a barrier.

3.5 Protocol

3.5.1 Owner of the protocol, governance

As an open standard, the protocol will be a common good, it can be used and implemented by whoever likes it. We will take care that the entry barriers are as low as possible. However, for some portions of the protocol a certification will be necessary, to reflect regulatory obligations and restrictions.

We propose a swiss based foundation as legal body, which formally holds the IP rights of the protocol and ensures that the protocol can be used freely.
We will establish a continuous, community driven protocol improvement process similar to the EIP process for the Ethereum Platform.

3.5.2 Outline of workflow elements of the protocol

- Application for policy
  Process of offering a product and applying
- Underwriting
  Process of accepting a policy
- Collection of premiums
  Payment process, one-time and regular payments
- Submitting of claims
  Process of submitting a claim, via oracle or manually
- Claims assessment
  Process of assessing a claim, via oracle or manually. A claims verification process allows the system to determine which policies are legitimately claimed and to propagate agreed payments to claimants. In the case of parametric insurance, this process references data feeds about insurable events and is (fully) automated.
- Identity Management & Privacy
  Process of KYC and AML, respecting privacy. This may involve private chains or off-chain storage of data.
- Admission / Certification
  Admission of participants to offer products and perform parts of the protocol
- Asset Management
  As funds flow in, we have to responsible use funds which are not immediately needed.

3.6 Community of customers, users and companies

The success of the platform will depend of a vivid community of users and companies. The token model should reflect and support this community. This community will play a central role in the realignment of incentives. Via tokens, customers can "own" their insurance. The community model should facilitate the development of future mutuals and P2P-Insurance models.
A community cannot be build from the outside, it has to grow from the inside. However, experience shows that there are some success criteria for communities. Famous open source pioneer Pieter Hintjens, http://hintjens.com/blog:10 has drafted some which we consider to be helpful for an in-depth discussion:

- Quality of mission
  A community can only grow pursuing a worthwhile goal. The goal must be super-individual and

- Freedom of access.
  The community should not have barriers or walls, it should welcome those of good will and encourage participation.
- Well-written rules.
  If rules are necessary, they should be carefully written and obvious.
- Strong neutral authority.
  To resolve conflicts, a strong but neutral authority should be in place, which can also be incorporated by some kind of governance mechanism.
- Proportional ownership.
  "You own what you make"

4 The DIP protocol token

The proposed protocol token is an integral part of a decentralized insurance platform. It will have some desirable properties: It will not introduce additional fees. The usage of the token is free, and owners of a token do not receive a revenue from the use of the platform. However, participants can use the token in their profit-seeking enterprises. Therefore, there is no incentive to fork the protocol, as you can't save costs by doing so.

4.1 Protocol & platform

The Protocol is a collection of Smart Contract Templates, Rulebooks, Standards, Best Practices which are developed and maintained by the community. There are many possible governance schemes for such a protocol; we intentionally don't make a prejudice on which model should be chosen, this will be part of the protocol development. The governance should fit to the participants using it. Of course, meanwhile blockchain offers some interesting tools to formalize governance, but that should be left to developers and users.
The platform is the community of all entities which make use of the protocol, and which are connected by a common economic interest.
Providing insurance is a complex process, involving possibly many participants, as we have seen above.

- Customers of an insurance need to rely on the smooth operation of these participants.
- Fees have to be distributed along the value chain, but only if all parts of a process have been appropriately fulfilled
- Participants supplying critical parts (e.g. a risk model) have to assume liability for their work.
- Some services are needed by many participants, so it makes sense to offer them as shared services.

The platform will serve as marketplace for insurance-related services, which are offered according to the open protocol standard and which are therefore always interoperable. Protocol relates to platform like chess rules to board & figures. For clarity, we will use the term "protocol" exclusively.

4.2 Role of the protocol token

To make long story short, we need some strong economic principles to ensure the proper working of all participants and their cooperative, mutually supportive behaviour. Therefore, we have designed the protocol tokens to bind participants to the platform and to assure the quality of the provided services. We are effectively implementing what is known as "Staking", focusing on the specifics of the risk transfer.

4.2.1 Example

Traditionally, economic relations are coded in form of legal contracts, which often have the form of “if-then” statements. “If you pay me $5.000, I’ll sell you my car”. In business contexts, we often have long-term contracts, like supply contracts or contracts for work and labor. These provide either a reward for a delivered good or service, a penalty for not delivering, or both.

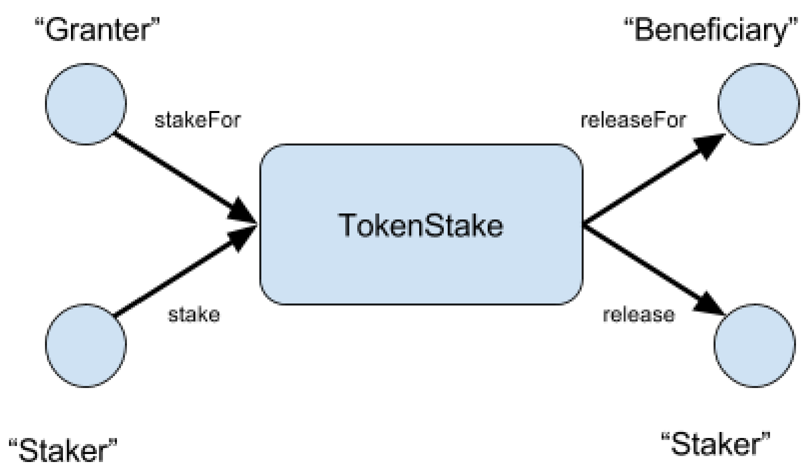
In our FlightDelay Insurance, we also have such a contract: we use oraclize to obtain provable data from our data provider, flight stats. Oraclize charges us with some finneys for calling their contract, but we have no guarantee that Oraclize will deliver. We have two options to incentivize Oraclize to provide their service properly:

1. in a “buyers market” (i.e. a market with many competing oracles) we could demand Oraclize to put some tokens in a “staking contract”, which will return the tokens if they deliver in time and forward the tokens to us in case they miss their obligations.
2. in a “sellers market” (i.e. a market with only one or few oracles) we can offer Oraclize an additional profit, again by staking tokens in a “staking contract”, but with reversed roles: Etherisc will stake tokens, and Oraclize will earn these tokens if they deliver, and in case they don’t deliver, the tokens are returned to us.
3. Of course, both options can be combined: both parties staking tokens, and the contractor earning tokens according to his performance.

The “staking contract” is very simple, it’s signature is

```
contract TokenStake {
[...]
  function stakeFor(address _staker, uint256 _value) public returns (bool);
  function stake(uint256 _value) public returns (bool);
  function releaseFor(address _beneficiary, uint _value) internal returns (bool);
  function release(uint _value) internal returns (bool);
}
```

The `stakeFor` and `stake` functions put some tokens in the contract where they are locked, until some predefined condition meets, in which case `releaseFor` or release are called and return the tokens to the respective `beneficiary`:

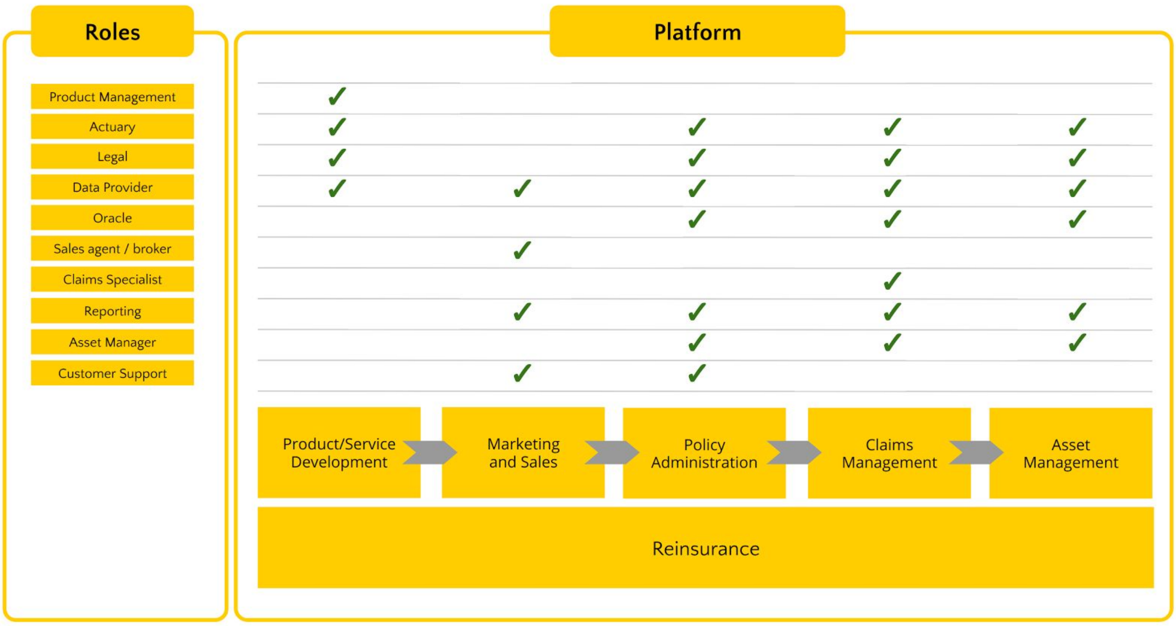


Example use cases for `stake`, `stakeFor`, `release`, `releaseFor`:

1. "`stake`": A contractor stakes tokens as collateral for providing a service at a certain quality / service level.
2. "`stakeFor`": A commissioner stakes tokens as reward for contractor, for providing a service at a certain quality / service level.
3. "`release`": Tokens are released to staker in case the condition is fulfilled / quality is proven / service level met.
4. "`releaseFor`": Tokens are returned to commissioner or "slashed" in case quality is not proven / service level not met.

4.2.2 Decomposing the value chain

Similar, a bunch of other contractual relationships could be modeled - each with a variant for "buyers" or "sellers" markets. . Thus, we would like to decompose the value chain as far as possible and to engage market mechanisms to select those participants which offer a service at the best value. The illustration shows the roles typically found in an insurance value chain, and which roles are needed for a particular step in the chain:



Basically, you can separate each role as an independent business, which can work together flexibly and bind themselves via reward-or-stake token contracts.
This is quite similar to the operating mode of a blockchain: Miners have an economic incentive for cooperative behaviour.
Some aspects of "good-behavior" comprise stability properties like:

- Promise to offer a certain service over a certain time (service stability)
- Promise to offer a certain service in a certain quality / with a certain SLA (quality stability)
- Promise to offer a certain service at a certain price (price stability)
- Promise to take a certain liability for a service (guarantees)

We propose to secure the platform and the products built on that platform via the protocol token. Participants (not customers) will need a certain amount of tokens to enter the platform "ecosystem". These tokens can be locked as collateral or offered as a

reward. Depending on the service offered, a different number of tokens will be required or offered to use the platform or provide services on the platform.. Simple services require a small number of tokens, complex or critical services will require a higher number of tokens. The amount of tokens which have to be provided as collateral or reward will correlate to the potential damage from participant misbehavior or from the violation of the platform terms. These parameters may be subject to a platform governance model (in the future) where participants have voting power based upon tokens owned. Or, governance may be conducted automatically by the use of smart contracts.
The proceeds from token sale(s) are used to nurture the development of the platform and to establish or provide central services as long as there is no independent participant providing them.
A certain insurance product needs a collection of services chained together to some business process. Participants offering these services can organize to offer such a product (maybe there is a market for such services and a “product management service” doing the coordination work). It is even possible that the fees for some of the services offered by participants in the ecosystem may be negotiated on an open-market platform. The protocol will offer ways to distribute the premium to the various risk pools and to the participants who provide product “processing”.

4.3 Participants on the platform and their use of the token

4.3.1 Customers

Customers can buy insurance using the token. For convenience, third parties can offer payment gateways and integrations which remove the necessity to own cryptocurrency from the end customer. Furthermore, participants can choose to offer insurance products in any native currency - be it a cryptocurrency, a token or a fiat currency.
Use of token: Universal currency to buy insurance products.

4.3.2 Risk Model Providers and Actuaries

Risk models are fundamental for any insurance product. The correctness of the model is precondition for the economic success of the product. With great impact comes great responsibility. Generally, because of the magnitude of value affected by errors and deviations in the model, a Risk Model Provider won’t take responsibility for the economic outcome of his model, but rather for his adherence to principles and established guidelines in his trade.
E.g. a risk model should be built on a clear specification, and it should be validated by acknowledged testing methods before it is put into production.
A risk model provider should therefore be rewarded according to such benchmarks. The economic risk of a model will usually stay with the party who runs the risk pool.
Use of token: Staking/Reward for providing or updating risk models.

4.3.3 Data providers and oracles

One of the most promising application of a decentralized insurance space is the way data is collected and managed. Customer data should remain in the control of the customer, and blockchain technology offers new ways of monetization of data.
Currently, data is collected together with the application for an insurance, and the insurance company “owns” the data - even after the insurance contract is no longer valid.
In a blockchain decentralized environment, the collection of data could be separated. Customers could get paid for voluntarily offering their data to a data pool, which in turn can sell this to interested parties, leaving the ownership of the data completely with the customer. Like a certificate in Keybase.io, data could be revoked at any time.
For the area of parametric insurance, oracles act as gateways to the physical world, providing provable and reliable ways to transfer data to smart contracts.
Use of token: Reward for giving data. Reward for giving access to data pools. Staking / Reward for providing reliable oracles.

4.3.4 Sales agents

Decentralized distribution will emerge as the blockchain space emerges as a whole. Sales agents can offer insurance products to business or end customers, receiving a share of the profit. The token can be used as a means to distribute revenue and profits among all parties involved in the production of a specific insurance product.
Use of token: Reward for distribution of products; means for distribution of revenue & profits.

4.3.5 Claims agents and Prediction markets

While the area of parametric insurance is rapidly growing in the ascent of IoT and the explosion of available data, there will remain many cases where an automatic detection and processing of claims is not possible, e.g. because the derivation of the loss from the event is to complicated or dependent on manual assessment.
Specialized and sometimes independent claims agents already exists e.g. in the area of car insurance, where they help insurers to process claims in shorter time.
These claims agents can immediately use a decentralized platform, as soon as adequate products are available.
For other use cases, prediction markets could be used to decide on certain relevant triggers for specialized insurance contracts, like cat bonds[13].
Use of token: Reward for the provided service. Incentive to start bids on a prediction market.

4.3.6 License providers

For the foreseeable future, insurance in most developed countries will depend on a proper license, which can be very difficult and costly to obtain. In some countries

[13] https://en.wikipedia.org/wiki/Catastrophe_bond

however, specialized companies offer “Protected Cell Company” model. In such a model, a license provider acts as an intermediary to regulators, bundling capital of many smaller projects to meet the minimum capital requirements.
Use of tokens: Staking tokens to provide capital for a license provider, paying fees for licenses.

4.3.7 Product Managers, Business Developers, Application builders.

Product managers and Business Developers scan the market for opportunities and orchestrate the necessary participants needed to build the actual product. Application builders can offer single products or even complete development kits where you can “your own insurance product” from some predefined templates.
Use of token: Reward for the provided service, fee for using application.

4.4 The DIP token: a protocol token

Etherisc is building a platform for decentralized insurance applications. Corporates, large and small, not-for-profit groups and insurtech startups can all come together to provide better products and services across the whole insurance value chain. We aim to use blockchain technology to help make the purchase and sale of insurance more efficient; enable lower operational costs; provide greater transparency into the industry compared to traditional operations; and democratize access to reinsurance instruments. Blockchain can provide the means to disintermediate the market with a peer-to-peer risk platform that helps insurance return to its roots as society’s safety net. We even envisage new groups building their own bespoke insurance risk pools and services on the platform. And Etherisc will be a fully-compliant, fully licensed insurance platform for the emerging blockchain economy.

The DIP token is a protocol token.

A protocol token (also known as an ‘Appcoin’ or ‘coin’) is an electronic asset that underlies a network. Tokens perform all kinds of functions depending on the network or platform they back, i.e. users use filecoin to store files on a distributed file-storage network and entities with open hard drive space earn filecoin for storing files. Tokens are an exciting new way to incentivize distributed networks and many uses have yet to be invented![14]

The DIP token is the building block for the emerging decentralized insurance economy on blockchain. Etherisc builds a decentralized insurance network which does not rely on an oligarchy of big parties, which control most of the business, like in the traditional insurance business. Instead, many participants can collaborate on new insurance products. Cooperation is welcome, also competition; but there won’t be artificial moats and barriers protecting some big players.
The notorious market entry barriers like high capital requirements and regulatory obligations are removed. The DIP token enables the economy on this network: participants in the decentralized insurance network cooperate in building insurance products. With the token, you can:

[14] From the CoinList FAQ: What is a protocol token?

- stake tokens as collateral
- buy insurance products
- interact with other participants to build decentralized insurance products
- pay the necessary fees and capitalization to obtain insurance licenses
- incentivize quality and proper behavior
- distribute revenues and profits among participants
- reward the provision of data by customers and participants
- pay oracles and prediction markets to resolve claims

The token itself does not add noteworthy friction to the network.
The token does not generate revenue or profit. The use of the protocol itself is free, the protocol open source, this is guaranteed by a swiss-based not-for-profit foundation, the DI Foundation. In a second step, separate "Risk Pool Tokens" will enable the capitalization of risk pools for specific insurance risks and will provide new instruments, which will further drive the adoption of the platform.

5 Tokenize Risk with Risk Pool Tokens

We propose a token model which enables participants to buy and trade the "long-tail" risks of a decentralized insurance portfolio and to gain exposure to its revenue as an income stream. Together with the consumer-facing insurance application, this forms a complete and fully functioning "trustless" insurance system on the blockchain. These risk tokens can be traded on an an end-to-end automated insurance and reinsurance marketplace. This platform will require no human intervention, and will be highly transparent to both customer and participant sides of the marketplace. We expect these tokens to be true "securities", because they will generate profits which are directly related to the managerial efforts of the creator, who provides the risk model for these tokens.
Due to the significant complexity of regulations we will build such a token system as a second step after establishing the core operational insurance business.
Nevertheless, we will give an outline of such a token system in the next sections.

5.1 General Concept

Conceptually, the platform has several components:

1. A **risk pool**, which holds a certain amount of reserve collateral used to issue and underwrite insurance policies against a predefined set of insurable events, within the framework of an **insurance model**.

2. A **reinsurance pool**, which holds extra collateral and reinsures the risk pool against catastrophic long-tail events which unexpectedly deplete the risk pool or render it unable to issue additional insurance.

3. A **risk management system**, which is a set of rules that governs the issuance, supply, inflation, and deflation of a digital token. For the FlightDelay Dapp (FDD)

we suggest to name the token RSC-FDD. Tokens are sold to collateralize the reinsurance pool and entitle holders to dividends from the risk pool's revenue stream.

4. A **token marketplace**, which allows participants to purchase and redeem tokens at economically fair and transparently calculated prices.

Under normal operation, the reinsurance pool holds a non-zero amount of collateral. The system is designed to constrict the total amount of risk underwritten to an amount no greater than the amount of collateral held by the reinsurance pool. At the outset, the reinsurance collateral is gathered through an offering of an initial fixed supply of RSC-FDD tokens (a crowdsale), and thus the upper bound of the number of policies that can be underwritten with 100% collateral backing is established. The system can be tuned toward a desired **maximum liability** level where the total risk of the insurance portfolio is capped considerate of market forces.

In turn, the risk pool automatically underwrites policies until this upper bound of policies is reached, and then ceases to underwrite policies. This is intended to ensure that every insurance policy is 100% collateralized and no customer can lose a payout to which she is entitled. (If this upper bound is reached but there is further demand for policies, the system's maximum liability parameter can be adjusted higher, and the system will automatically issue and sell tokens to support new policies with minimal dilution.) Also note that a $1M capitalization of the reinsurance pool will support a vastly larger throughput of policies than will likely be required in the early stages of the project.

To support normal operation, a minimal collateral reserve is required to be held in the risk pool, and this value is determined by the insurance model. Insurance premiums are calculated as a function of this required collateral, the insurable event in question, and the desired payout for the policy at claim time. The exact calculation is specific to the model, but note that the risk pool is able to subsidize premiums by reserving excess collateral through a variety of means, such as seeding the pool with initial auxiliary capital or retaining revenue in the risk pool.

At the time a customer purchases a premium, a 5-10% fixed fee will be assessed on the premium and allocated toward operational costs.

5.2 Calculating the required capital

The primary concern of any insurance model is to calculate the reserve capital required to guarantee solvency of the risk pool to some arbitrary and high confidence level, such as 99.99%.[15] Under normal circumstances this results in an automated system where risk is shared among policy holders. Since the actual collateralization of the risk pool is

[15] Note that in many jurisdictions, regulations require lower confidence levels. For instance, in the EU Solvency II requires a “a confidence level of 99.5% over a one-year period”. See http://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32009L0138 Article 101.

usually higher than the actual number of claims that must be paid, the risk pool has a positive probability expectation of revenue. When a policy expires without a claim, its premium becomes revenue and it is allocated as follows:[16]

1. 10% is reserved in the risk pool to subsidize premiums.
2. 20% is paid to the reinsurance pool to subsidize long-tail risk collateral
3. 70% is paid pro rata to the holders of RSC-FDD tokens as dividends.

In exceptional circumstances, an outsized number of policies are claimed and this can result in depleting and exceeding the collateral reserved in the risk pool. In this case, the claim liability is paid out to customers from the reinsurance pool, whose precise function is to service this long-tail risk.

An event which depletes the reinsurance pool in this way results in a level of collateral below the targeted liability level desired by the business, and the system will issue new RSC-FDD tokens in order to replenish the pool accordingly. The reinsurance pool is also replenished through the revenue flow described above, and tokens are automatically purchased back from the RSC-FDD token marketplace when the reinsurance capital exceeds the targeted capitalization. This, in turn, results in deflation of the RSC-FDD token supply (or an increase in potential acceptable business risk liability) and a token supply which remains “managed”, increasing only at the rate by which the business is able to increase its throughput of policies underwritten.

This proposed economics has several desirable properties:

1. **Solvency Guarantees.** No customer can lose money as insurance policies are underwritten against 100% collateralization in the risk and reinsurance pools. No insurance policy will ever be issued that is not fully backed by collateral.

2. **Natural Scalability.** If the demand for policies exceeds the available collateralization, the system has a natural mechanism to scale up to meet the desired demand through additional RSC-FDD token issuance. In the same way, it can naturally scale down to adjust to decreasing demand.

3. **Fair Token Pricing.** The fair price of tokens is transparent, as it is the present perpetuity value of a measurable dividend stream which is itself well-defined by the probability model of the insurance portfolio. Given a reasonable risk-free rate and the observed recurring revenue stream of the risk pool, the price of tokens can estimated without resorting to speculative markets for pricing.

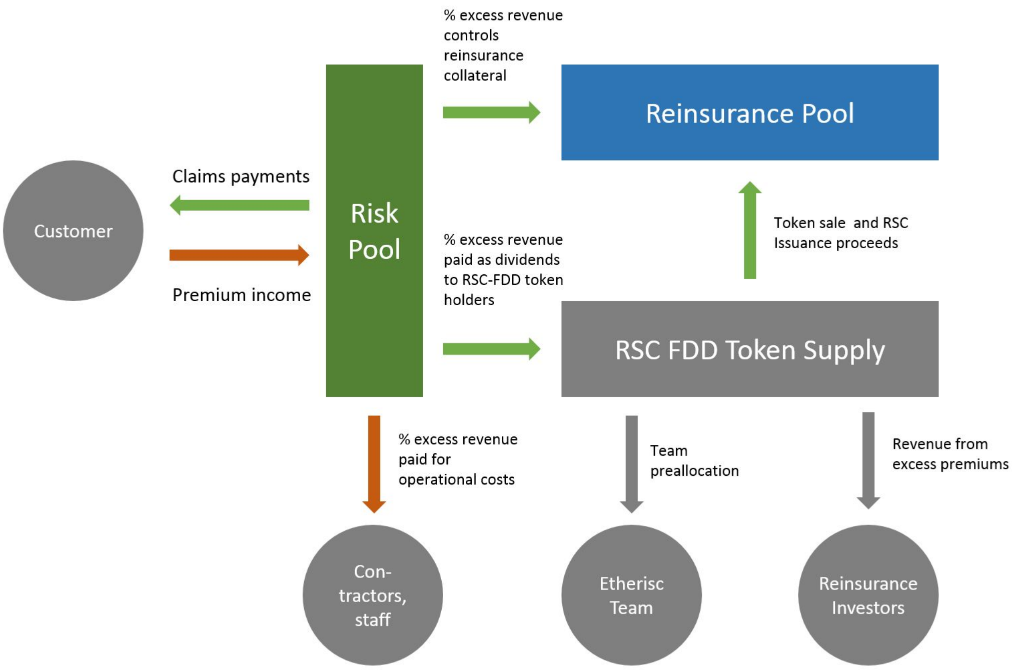
4. **Value Proposition for Crowdsale Participants.** Under reasonable risk-free rates available on cryptocurrency-focused markets, such as the Poloniex BTC

[16] Note that the following allocations are subject to change at any time prior to launch based on new modeling.

lending rate[17], and assuming modest utilization of the proposed insurance product, the fair pricing of tokens results in substantial incentives for crowdsale participants.

5. **Low Dilution.** Under reasonable risk-free rate assumptions and even modest utilization of the proposed scheme, dilution is likely to be low. This is due to the fact that tokens gain a substantial increase in value after an end-to-end beta product has been delivered without exceptional occurrences and expects a non-zero future revenue stream.

The following diagram outlines the components and value flows of the proposed system.



5.3 Risk Management for Flight Delay Insurance

An insurance risk is a future liability for payment which occurs with a certain probability. In the case of the flight delay insurance, the probability is calculated from historical flight data. Such data is readily available from multiple trusted sources.[18] We assume the data quality for flight delay data to be sufficiently accurate for our purposes, considering that airlines are legally obligated to provide such data and it is highly audited by various businesses which rely on it. Insurance risks associated with flight delays are ideal for a proof of concept product, as individual premiums represent very

[17] See https://www.poloniex.com/lending#BTC and https://cryptolend.net/rates.html for current and historic rates.
[18] For instance, FlightStats (http://flightstats.com).

small financial risks to consumers and participants and under normal circumstances flight delays are well-approximated by independent probability events.

A simple algorithm for premium calculation is that the average payout for claims is covered by the net premiums collected[19]. This works well in most cases, but there can be periods where - due to statistical fluctuations - the payout is higher, sometimes even much higher than the average. This is compensated by other periods in which the payout is lower, but we have to provide enough funds that under “most circumstances” all payouts can be made. “Most circumstances” can be refined to a “level of confidence”: a level of confidence of e.g. 99.9% means that in 99.9% of all distinct periods of time the probabilistic gross payout is smaller than the sum held in the risk pool.

In our model, we calculate with an even higher level of confidence of 99.99%. Put into simple words, that means that in the average, for every 10,000 years there will be one year in which the risk pool doesn’t have enough funds to pay all claims. Such a level of confidence exceeds the legal requirements e.g. in Europe by far - the “Solvency II” regulations require only a level of 99.5% probabilistic reserve.

Providing a risk pool at such a high level of confidence comes at a cost. Participants have to provide the funds and expect a fair return for taking the risk and binding their capital. This return is paid from excess premiums, which were not needed to pay claims, and in the end the revenue from the risk pool is designed to provide a certain surplus in excess of the net premiums collected to pay the average expected claims.

5.4 Target Parameters of the Risk Model

Participants will ask for the key parameters of our model, therefore we provide some estimates which we will elaborate more precisely in the future. Note that all values are subject to change.

Parameter	**Estimate**
Risk pool solvency confidence level	99.99%
Fixed service fee on premiums	5-10%
Target return rate on reinsurance pool	5-10%
Target maximum liability at launch	$1M max
Target policy throughput	2000 concurrent policies @ $500 average payout

[19] Net premiums are the premiums paid by the customer net of any fixed service fees.

6 Etherisc the company

6.1 Business Plan

A business plan for our first product, the FlightDelay Insurance, is available as separate Document.

6.2 Team

6.2.1 Founders

Christoph Mussenbrock
Christoph has a long record of accomplishment in the cooperative banking sector in Germany. After several years on the board of a cooperative bank, he switched to the IT segment and became Chief Program Manager Credit Solutions and Chief of Strategy Development at Fiducia & GAD IT AG – one of Germany's biggest IT Service Providers. Since 2015, he has been CEO of parcIT GmbH, one of Germany's best-known companies specialized in risk management solutions.

Due to his many years of working in the field of banking and insurance, Christoph is highly experienced in all matters concerning regulatory frameworks. He also co-founded Progeno Wohnungsgenossenschaft eG, a housing cooperative in Munich, which has successfully crowdfunded a large residential project. Christoph has a master's degree in mathematics and wrote his thesis on formal soft- and hardware verification.

Stephan Karpischek
Stephan has more than 20 years experience in IT businesses and advises finance and telecom enterprises on digital strategy. In 2015, he was part of the of the UBS crypto 2.0 innovation lab at Level39 in London, applying blockchain to banking use cases.

Stephan has been involved with digital currencies since 2008 and has a PhD in information management from Swiss federal technical university ETH Zürich. He wrote his thesis on mobile applications for the Internet of Things.

Renat Khasanshyn
Before joining Etherisc, Renat was Venture Partner at Runa Capital and CEO of Altoros. Most recently, Renat led from its inception, the insurance practice at Altoros together with its key customers Allianz, Allstate and Liberty Mutual, focusing on canonical use cases of blockchain, reinsurance and insurance securitization (catastrophe bonds). Renat co-founded Altoros, a 250+ employee strong software and research laboratory in the area of distributed databases, container orchestration & developer marketplaces.

Renat began his career in 2001 as a software engineer at a regulated insurance intermediary in Tampa, Florida. Using Perl/CGI/LAMP, he built one of the first online distribution portals in the insurance industry. Real-time quoting, payment and policy issuance gave uninsured consumers in the US same-day access to a network of 30,000 doctors. In 2007, Renat co-authored Apatar, - GPL-licensed, 100% open source data integration tool, and co-founded Belarusian Java User Group. Renat studied Engineering at Belarusian National Technical University.

6.2.2 Advisors

Ron Bernstein
Ron is the CEO of AugmentPartners Limited, a private software development company focused on decentralized trading dApps, market liquidity, and blockchain data management. Ron is also an early advisor to the Augur Project -- a decentralized Prediction Market built on the Ethereum blockchain.
Previously, Ron founded Intrade.com and Tradesports.com. Intrade was the world's most popular (centralized) Prediction Market from 2007 until 2012. Prior to focusing his attention of crypto assets, Ron traded commodity options and derivatives for more than 25 Years on the trading floors in NYC and London.

Ralf Glabischnig
Entrepreneurs. Invested. Involved. This is the core mindset that Ralf embodies as Co-Founder of Lakeside Partners, a leading early-stage investment company in Crypto Valley. With 20 years of experience as a business- and IT-consultant and in his role as Managing Partner at inacta AG, a major Swiss Information Management solution provider, he possesses extensive expertise in transforming the insurance industry, as well as a diverse entrepreneurial background stemming from several ventures and advisory board positions. Ralf brings a passion for innovation and first-hand knowledge of the Swiss business landscape in his role as strategic advisor to the Etherisc project.

William Mougayar
William is a Toronto-based investor, researcher, blogger, and author of The Business Blockchain (Wiley, 2016). He is a direct participant in the crypto-technology market, currently on the Board of Directors of OB1, the OpenBazaar open source protocol that is pioneering decentralized peer-to-peer commerce, a former Board Advisor to the Ethereum Foundation, board member at Stratumn, a member of OMERS Ventures Board of Advisors, an Advisory Board member to the Coin Center, Bloq and other leading blockchain organizations. He blogs regularly about the present and future of blockchains at Startup Management.

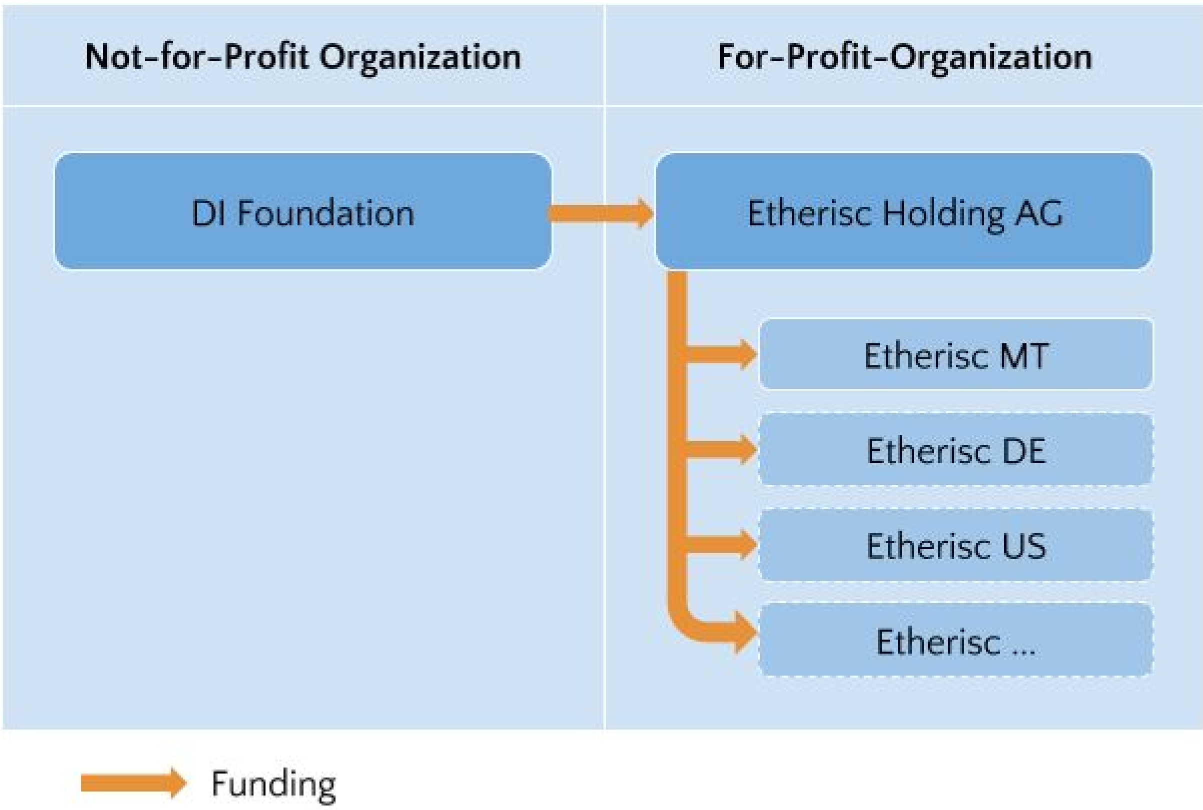
Jake Brukhman
Jake is Co-Founder and Managing Partner at CoinFund LLC, a blockchain advisory company and a cryptofund operating since July of 2015. Jake has 9 years of experience in pure and financial technology, a background in computer science and mathematics, and an avid interest in blockchain and financial technology. He is co-author on multiple blockchain whitepapers including Etherisc, Kin, Sweetbridge, and others. Previously, Jake was Partner & CTO at Triton Research, a technical product manager and engineer at Amazon.com, and spent five years as a financial technologist at Highbridge Capital Management and as a quantitative researcher at Kohera.

Tobias Noack
Since 1991, insurance broker with ATS Insurance Brokers. Subsequently, power of attorney and shareholder role. Acquisition by Aon Risk Solutions (ARS) in 2004. Since then held positions as Regional Manager for ARS Berlin (Germany East), as well as in Key Accounting and Sales. Member of the ARS Operational Board. Since 2016, Special Projects role.

6.3 Legal & Regulatory Strategy

6.3.1 Legal structure

One of the core concepts of the Etherisc Approach is the “Two-Folded Approach” with an independent, not-for-profit foundation (the “DI-Foundation”) and commercial entity/entities (bundled in an “Etherisc Holding AG”).



For the success of the token sale, but also for the success of the vision, and last but not least for the legal setup, it is crucial to have a good understanding of the “Why” of this construction.

First, we believe in decentralization as one of the core concepts of blockchain. As Fred Ehrsam states: “So the biggest question I ask myself when evaluating a blockchain idea is: "is this uniquely enabled by the new paradigm?"
https://twitter.com/FEhrsam/status/902584358770434048

To be able to execute this statement, we need the power to do so. Clearly, the success of any decentralized organization cannot be enforced by central entities, but to avoid the "Tragedy of the Commons", central entities can help, if they are clearly dedicated to the common goal.

(again, a recent tweet from Fred Ehrsam: "As a result, Ethereum is starting to suffer from a tragedy of the commons problem: while lots of people own ETH and would benefit from Ethereum improving, the economic reward for any single individual improving it is low." https://twitter.com/FEhrsam/status/900745083426791425)

The one and only reason for our two-folded approach is the attempt to give an answer to this question. Now, what's the answer?

The Tragedy of the Commons is a Di-Lemma. On one side, you need common goods which are accessible without fees by everyone (of course there are some rules of the game, but no fees, and no profit). On the other side, the common goods need to be nurtured which comes at a cost.

Our solution: First, a decentralized foundation, which is strictly neutral and has only one purpose: to develop the open platform and keep it open for all times.
Second, a commercial entity which acts as first mover on the platform, shows its feasibility and earns money which will ensure the sustainable development of the platform.

Now that we have solved the tragedy of the commons from the start, we have to ensure that this will endure. The Foundation is bound by its codified purpose, which is "eternal" (cannot be changed easily). But it its a "lame duck" by purpose - it has to stay neutral, and its funds are limited. Provided the foundation is equipped with enough funds from the beginning, this can be tolerated, but it will inevitably come up again after some time.

The commercial entity is more difficult. The owners can always change its purpose, the direction of its commercial activities and the use of its profits. There is only one way to ensure that this commercial entity is bound "forever" to the purpose we give it at the start: at least a blocking minority has to be under control of an entity which itself is bound - which is very naturally the Foundation. This also solves the problem of the foundation mentioned above, having no natural income otherwise.

Having a blocking minority in the commercial entity has a second advantage. The foundation can take care that the commercial entity is always serving the open platform and not cannibalizing it. The commercial entity will thus act as a Shared Service Provider for the open platform.
Especially, in our case of a decentralized insurance platform, the commercial entity will reduce the regulatory moat for all participants, offering a simple access to licensed insurance products and structures, in a way which could be similar to the Malta "Protected Cell" structures.

Last tweet of Fred Ehrsam on this: “Ideas uniquely enabled by a new paradigm have never been seen before. So we have to remix current ideas to get there.”
https://twitter.com/FEhrsam/status/902583309972103168

If you ask “Can this work?”, there is one big example for such a construction, which is very successful: the German Cooperative Banking System. The cooperative banks where a rock of stability in the 2008 financial crisis, and they have exactly this setup.

All these banks are legally independent, but working in close cooperation. They have an not-for-profit association, which per definition is neutral regarding commercial strategies, but helps to formulate and cultivate such strategies. E.g. they do a lot of market research, organize larger projects.
More important, they have common Shared Service Companies, which perform tasks a single cooperative bank is not able to perform itself, like international payments, IT Operations and the very german concept of a “cooperative building society” (Bausparkasse). Even an insurance is part of the system, and it is one of the biggest. The whole cooperative sector has 18 million members, and a market share of > 30%. The Shared service companies are always among the biggest in the respective market.

There is one important caveat: While this is very stable, there is a natural - but creative - tension between the Shared Service Companies and the cooperative banks, because naturally, the Shared Service Companies seek their opportunities and this can lead to a conflict of interest. Therefore, a very clear governance is needed, and of course a strong and common vision.

6.3.2 General regulatory strategy

Etherisc aims to enable fully-licensed and fully-regulated insurance products on its decentralized insurance platform. To achieve this ambitious goal, we have been in contact with regulators in multiple jurisdictions to educate them on the role and benefits of blockchain technology in the insurance space. We strongly believe regulatory safety is an essential component of a decentralized insurance platform, and we are working with both regulators and insurance partners in the major markets to be able to roll out commercial products. Acquiring proper authorizations in every market where we will be selling insurance is critical, and we expect to be authorized as an insurance company in one of our top target markets sometime between Q4 2017 and Q2 2018.

Obtaining authorization to underwrite insurance is a collaborative process involving multiple partners and specialized service providers. Earlier this year, we initiated an evaluation of insurance management companies to support our application for authorization. In July 2017, the team selected a service provider in the EU and began the process of authorization.

We also plan to share insurance licenses with other insurtech startups as licensing will be one of the services that the platform provides.

Our regulation and licensing plan is a multi-step process: For the first stage, until we have our own license, we will use insurance partners to sell insurance products. We will also apply for our own insurance license, and engage a reinsurance partner to sell insurance products. Our longer-term vision is to replace reinsurance with a decentralized insurance market.

6.3.3 Approach in Malta

We are partnering with Malta-based Atlas Insurance Malta. The Atlas Group of Companies forms one of Malta's foremost insurance and financial services organisations with a staff complement of over 170. The flagship company of the Group is Atlas Insurance PCC Limited ("Atlas"), a general business insurer operating an extensive intermediary network across the islands.

Atlas was the first direct insurer to convert to a Protected Cell Company (PCC). Atlas gives promoters the opportunity to own their own EU insurance vehicle with less capital and less cost, also avoiding fronting requirements. Atlas is an independent PCC giving the option to promoters to outsource cell management to authorised insurance managers.

Cells in Atlas can also write third party risks, where our substantial active core provides added security and flexibility. The company maintains substantial surplus funds over regulatory requirements.

Starting in October 2017, Atlas provides us with capacity to start a small pilot for selling FlightDelay insurance. In 2018, we will formally establish a Protected Cell entity to explore the "Protected Cell" model as role-model for decentralized insurance.

6.3.4 Approach in the UK

In the UK, we have established Etherisc Ltd. as a legal body to apply for a FCA sandbox license. Extensive talks with the FCA, the PRA and our insurance partner in the UK have led to a preliminary acceptance of Etherisc Ltd. to the sandbox program in early 2017. We plan to become part of the third cohort of companies to obtain a license to sell insurance products in the UK.

6.4 Risk Management

Etherisc will establish a risk management system in the DI Foundation as well as in the commercial entities, bundled in the Etherisc Holding AG.
We have inventoried the main risks which come along with the disruption of a century old business in a completely new technological context. The risk monitoring and managing process will implement the requirements of the respective jurisdictions.
The expertise for setting up a risk management system which fulfills the regulatory requirements is available in the founders' team.

7 Token Sale

7.1 Token Sale Structure and Timeline

The terms & conditions of the token sale as well as the exact timeline will be described in a separate document, which will be published at the time of the announcement of the token sale.

7.2 Meeting capital requirements

The funds collected from the token sale will be transferred in a swiss-based DI Foundation. This transfer - which is, technically speaking, a donation - will enable the DI Foundation to accomplish its goals, which are hard-coded in its purpose. The “Eidgenössische Stiftungsaufsicht” will then supervise the operations of the DI Foundation and enforce the execution of the Foundation's’ purpose.
While the foundation will keep its own business lean and cost-effective, it will use its funds in two main areas:

1. Financing the development of the Decentralized Insurance Protocol and the community of customers, users and participants
2. Establishing commercial insurance companies and providing the capitalization for these companies, either alone or together with other partners, preferably from the insurance business.

We can deduce the funding goals for our token sale from these two fields, which require and empower each other. Of course, there is a large bandwidth for these numbers. In what follows, we will give lower and upper bounds together with indications how a higher funding will enable a broader or faster approach to our overall objectives. The dependency of the scope from the achieved funding is different for the two areas mentioned above.

For the first field - the development costs - and for simplicity and better understanding, we can organize our estimates in “Levels” (Bronze, Silver, Gold, Platinum) with the following meaning:

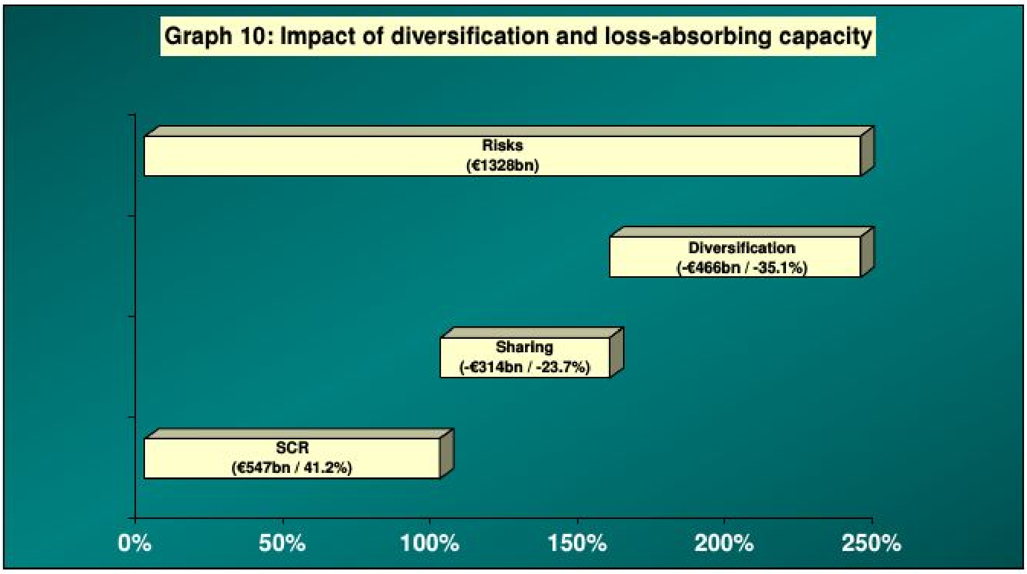
Level	Name	Description	Cost USD	FTE
1	Bronze	Accomplishing basic objectives. Basic protocol components. Deliver single working parametric product. Achieve self-sustainability. Regulated entities in cooperations e.g. dependent PCCs	5.0 M	15

2	Silver	Accomplish basic + some advanced objectives: Full protocol components. Deliver several working parametric products. Achieve revenue stream for foundation. Regulated entities in cooperations e.g. dependent PCCs	7.0 M	25
3	Gold	Fully accomplish all objectives. Full protocol. Gateways & interaction with other protocols. Deliver first example of non-parametric product. Achieve revenue stream for foundation. Independent primary insurer.	9.0 M	30
4	Platinum	Extended operations: Full protocol & gateways. Risk trading platform with Risk Pool Tokens. Broad palette of products parametric / non parametric. Revenue stream for foundation. Independent primary insurer and reinsurer.	11.0 M	35

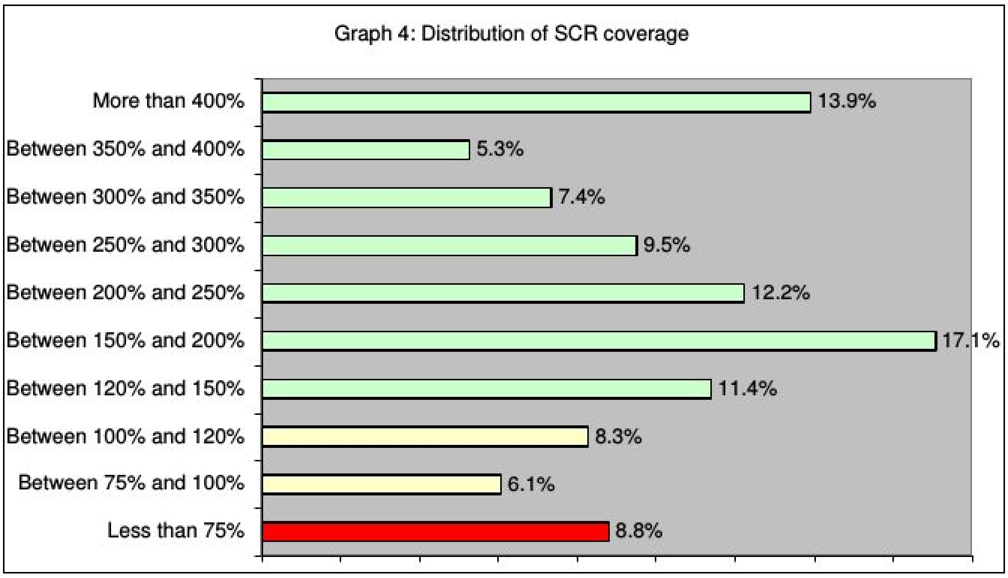
The funds will be spend over a period of 2 years. We calculate a FTE (Full Time Equivalent) with average cost of USD 100K per year and a relation of HR to other costs of 2:1. The costs will be distributed between DI Foundation and the commercial entities in a ratio of roughly 1:1.

For the second field - the capitalization of commercial entities, which act as subsidiaries of the foundation, the metric for the capitalization is different. Due to the basic regulatory approach the business volume of an insurance company is roughly proportional to its capitalization, because in insurance, business volume means risks, and risks need to be covered by own funds. To get an estimate, we can examine some quantitative material, e.g. the results of the famous “Quantitative Impact Study 5”[20] which was conducted prior to the introduction of Solvency II. It gives a very rough estimate for the ratio between the “SCR” - the Solvency Capital Requirement - and the sum of individual risks taken by an insurance company.

[20] https://eiopa.europa.eu/publications/reports/qis5_report_final.pdf



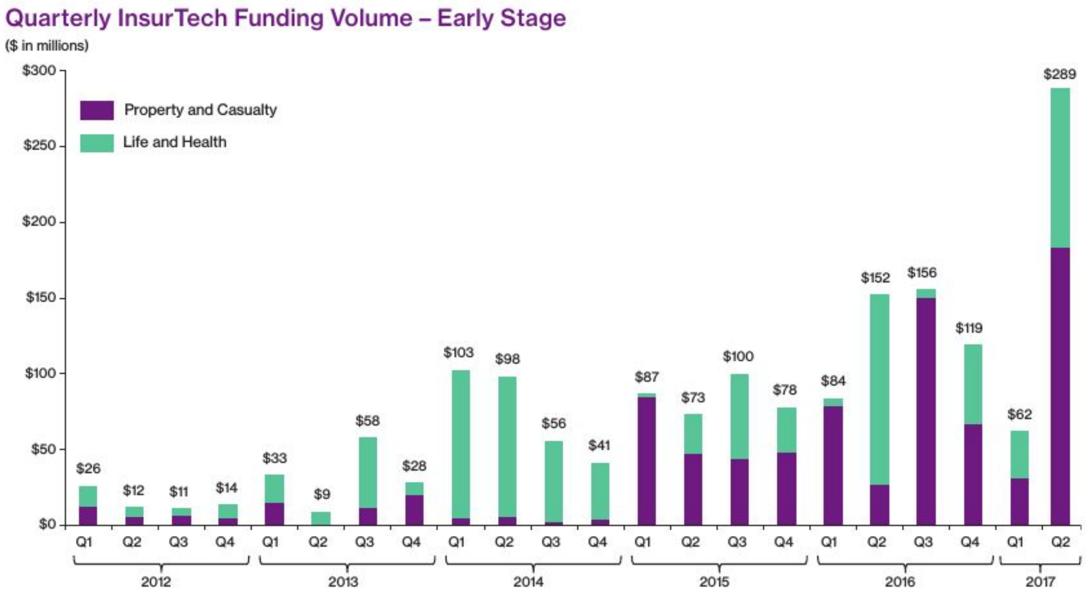
From the graphic, we learn that for € 1M insured risks we need a minimum of about 41.2% = € 412K own funds to fulfill the Solvency II capital requirements. The remaining risk is covered by diversification (35.1%) and sharing (23.7%).
This number, however, is a minimum - most insurance company grossly overachieve this by factors up to more than 400%:



Furthermore, due to the operational complexity of an insurance business, an insurance company is difficult to operate at the bare minimum of capital. Economies of scale become effective at larger scales, and therefore we estimate the minimum solvency capital required to run a sustainable insurance at about USD 10M.
Solvency capital is, however, only one part of the equation. Regulations in most countries require the provision of a separate "organizational fund" to finance the operational costs of ramping up an insurance business. The organizational fund is typically between 25-50% of the necessary solvency capital.
In total, we can estimate the minimum total capital requirement for starting an insurance business between USD 12,5M - USD 15M.

This number is plausible, because its in the same magnitude as the bare minimum capital required to establish an insurance company in Germany and Switzerland, and it corresponds as well to the capitalization of some newly founded insurance startups in Germany[21].

However, the same report shows that most promising startups are much better capitalized. Lemonade e.g. commands USD 60M after their third round, trov has USD 84 M and indian based Digit Insurance has another USD 60M.



Deal Count

P&C:	4	1	4	3	4	2	5	5	4	2	5	6	8	8	8	13	33	9	23	20	15	21
L&H:	6	3	6	7	8	4	6	4	7	9	16	8	7	8	10	18	8	9	4	11	10	19

7.3 Deduction of token sale range

From what has been said, we have a range of USD 5-10M for development costs and a starting point for capitalization of USD 12.5 - 15M.
Coming to Etherisc, we have several additional factors to take into consideration:

1. Different from the startups mentioned above, which are typically financed over various rounds, Etherisc will need to capitalize in only one round.
2. Etherisc is operating in a completely new field, with less mature systems, with unproven economics. We are pioneers in every aspect of our model, and for this systemic risk an adequate capitalization is mandatory.
3. For the same reason, we expect that regulators will demand higher capital requirements than an equivalent insurance operating in a traditional model.
4. Ramping up our business can last longer than expected, and it is not yet clear how fast mass adoption of crypto currencies and crypto business models will start.

[21] http://i3analytics.com/wp-content/uploads/2017/07/CB-Insights_Insurance-Tech-Q2-2017.pdf

We therefore target a hard cap of USD 50M, but we can as well take off with as much as USD 20M. Reaching the hard cap would give us the runway to develop protocol and community more organically and would significantly reduce risks.
But for every amount north of the minimum we can start as well.

7.4 Protection of Participants and Transparency

Etherisc is dedicated to a high degree of transparency, as long as legitimate interests of participants, customers and employees are taken into account. To work not only on Etheriscs own transparency policy, but also enhance the transparency of the whole blockchain sector, Etherisc has joint “Project Transparency”[22], a joint effort of some of the best reknown projects in the space, including Aragon, Lykke, Maecenas, to name only a few.
Etherisc and the other members of Project Transparency follow a self-inflicted policy of making public the purpose of every use of funds which exceeds 0.5% of collected funds.

7.5 Migration of RSC Tokens

We use RSC-DST to denominate the RSC tokens which are issued in the DST contract of hack.ether.camp, and DIP for the future tokens which are issued to finance development of the decentralized insurance protocol.

Proceeds from selling RSC-DST tokens have been used to back research and development and initial operational costs. RSC-DST token holders will receive a fair share of DIP-tokens in the upcoming “Token Generating Event” (TGE) for their engagement in an early stage of Etherisc. The conditions will be published together with the tokensale document mentioned in chapter 7.1.

7.6 Token Sale contract and audits

The token sale smart contracts have been written by the team. The code has undergone three independent audits of well-known solidity experts, which will be published as soon as the final version of the contract is considered stable.

[22] www.projecttransparency.org

8 Appendix

8.1 Example application to the use of an oracle in an insurance context

The TokenStake contract serves as an abstract interface. It provides four functions: two functions for staking and two functions for releasing: [23]

```
contract TokenStake {

  function stakeFor(address _staker, uint256 _value) public returns (bool) ;
  function stake(uint256 _value) internal returns (bool);
  function releaseFor(address _beneficiary, uint _value) internal returns (bool);
  function release(uint _value) internal returns (bool);

}
```

The `staking` functions are public, anyone can transfer tokens to the contract. To notify the contract about the incoming tokens, this has to be done in two steps:

1. The token owner `approves` the TokenStake contract over the sum of tokens to be staked.
2. The token owner calls `stake` or `stakeFor`. The TokenStake contract then `transfers` the token to the TokenStake contract and records the tokens in an internal ledger.

The release functions are internal, therefore the TokenStake contract itself has not much utility: If you `transfer` tokens to the original Tokenstake contract, the tokens will be locked forever.

The TokenStake contract becomes useful if you extend it with some additional logic, which binds the `release` functions to some condition. This condition can be arbitrarily implemented. A typical use would be a time lock: the `release` function can be called after a certain block or time is reached.

As a more complex example, we present another use case where the TokenStake contract is used to reward an oracle for providing the information in due time. First, we present the basic skeleton of a blockchain oracle:

```
contract usingOracle {

    event Query(string _request, bytes32 _id);
    uint256 count;

    function query(string _request) internal returns (uint256) {
        count += 1;
        Query(string _request, id);
        return count;
    }
```

[23] we show only the function signature, the complete code can be found here: https://github.com/etherisc/tokensale/blob/develop/contracts/protocol/TokenStake.sol

```
    function __callback(string _result, uint256 _id) public {
    }
}
```

A contract using this oracle will extend this interface. It will then call `query()` with a string containing the actual request (e.g. a http url with an api call, or an sql query). The actual oracle will watch the `Query` event, perform the query offchain, and then call `__callback` with the result. The `id` parameter is used to discriminate parallel calls to the oracle.

We can now extend this basic interface with a reward mechanism: the oracle receives 10 token for each `__callback` which is performed within 30 minutes of the respective query:

```
contract usingIncentivizedOracle is usingOracle, TokenStake {

    mapping(uint256 => uint256) public deadlines;

    uint256 deadline constant = 30 minutes;
    address constant contractor = 0x1234abcd1234abcd1234abcd1234abcd1234abcd;
    uint256 constant reward = 10;

    function query(string _request) internal returns (bytes32 _id) {

        _id = super.oraclize_query(_request);
        deadlines[_id] = now + deadline;
        return _id;
    }

    function __callback(bytes32 id, string _result) public {
        if (now < deadlines[id]) {
            releaseFor(contractor, reward);
        }
    }

}
```

Finally, we put everything together in a simple insurance contract. We assume that somebody has already approved the insurance for the transfer of a number of tokens. In case of a timely delivery of the request, the oracle (with address `contractor`) receives 10 tokens.
The oracle - simplified - checks the “flight” and yields “ok” if the flight can be insured:

```
contract Insurance is usingIncentivizedProxy {

    function Insurance (uint256 _stakedTokens) {
        stake(_stakedTokens);
    }

    function newPolicy(string _flight) public {
        id = query(_flight);
```

```
        // store policy, … perform further checks ...
    }

    function __callback(uint256 _id, string _result) public {
        super.__callback(_id, _result);
        if (_result == 'ok') {
             underwrite(_id, _result);
             // notify customer that the policy has been accepted.
        }
    }
}
```

8.2 Credit Risk Model

8.2.1 Abstract

This is a technical addendum to the Etherisc[1] decentralized insurance whitepaper, presented at the EtherCamp Virtual Accelerator (http://hack.ether.camp). In this work, we develop a simple probability model for a parametric insurance pool with variable claim payouts and variable probabilities of insurable events. We present one methodology for calculating premiums. We also suggest avenues for employing this model in a real-time context of a portfolio that continuously underwrites claims. Finally, we present a Python simulation which puts our modeling methodology into practice against actual data for insurable flight delays.

8.2.2 Overview

In this section, we describe the desirable properties of a basic credit risk model for decentralized insurance with a minimum viable set of features that can enable the proof of concept Etherisc application. In general, insurance works by selling policies which cost the purchaser a premium in exchange to an entitlement to a payout in the case of some *insurable event*. The occurrence of the event and subsequent automated payout is referred to as a *claim*. The Etherisc demo application focuses on the case of flight delays, but in our model we do not make any assumption about the specific nature of the event with an eye to expanding our insurance offering to different markets.

An insurance risk pool underwrites policies by taking on some maximum liability in the form of credit for potential claims, and collateralizing the portfolio with a smaller amount of capital to cover the capital outflow due to claims with a reasonable confidence level. If the portfolio experiences a capital outflow of claims which exceeds the collateralization of the risk pool, the risk pool becomes insolvent. This problem of excess risk management is addressed by the Etherisc whitepaper and its decentralized reinsurance market based on cryptographic tokens.

We target the following mathematical properties in our model of an insurance risk pool:

1. The portfolio should be able to reasonably model insurable events as independent (and uncorrelated) random variables.
2. The model should be able to provide distinct probabilities of each individual event as parameters.
3. The portfolio should be able to underwrite policies for an arbitrary payout amount.

[1] http://etherisc.com

4. The model should be able to parametrize the probability of solvency of the portfolio at an arbitrarily high confidence level.

In modeling insurance, correlation analysis of events plays a key role in achieving highly realistic models but significantly increases model complexity. For the purposes of our proof of concept product, we assume independence of insurable events and offload the risk due to error to our reinsurance market (covered in the Etherisc whitepaper). We recognize the importance of correlation analysis and future efforts will focus on developing more granular modeling methodologies. Note that this model's ability to underwrite payouts of arbitrary size is crucial because it enables the risk pool underwrite multiple policies pertaining to the same event without requiring a model based on correlated random variables.

In general, we also desire that our model outputs should cover the gamut of relevant financials that might be useful in constructing a smart contract[2] which manages the insurance pool: (i) the total liability of the credit portfolio; (ii) the required collateralization of the risk portfolio; (iii) at least one plausible method for calculating premiums commensurate with event probabilities and payouts; and (iv) an expectation and spread for the revenue of the portfolio (ideally, non-negative).

Finally, our model should be straightforward to calculate (or estimate). In the following section, we develop the mathematics for a model which conforms to the properties above.

8.2.3 Probability model

Let us assume that our risk pool portfolio contains n policies which are insuring against n insurable events, modeled as independent but not necessarily identically distributed random variables X_i $(i = 1, \ldots, n)$. Suppose that P_i^* $(i = 1, \ldots, n)$ is a fixed set of desired payouts where P_i^* corresponds to the policy i. Furthermore, suppose that $X_i \in \{0, 1\}$ are Bernoulli random variables with event probability p_i $(i = 1, \ldots, n)$. That is, $P(X_i = 1) = 1 - P(X_i = 0) = p_i$ for $i = 1, \ldots, n$.

The *total liability* L of the portfolio is the sum of all payouts the portfolio is underwriting, and we define

$$L(n) := \sum_{i=1}^{n} P_i^*.$$

X is a random variable that has a non-trivial distribution which is the weighted sum of Bernoulli variables with non-uniform probabilities. Let us define π to be the desired confidence level (probability) of portfolio solvency, and we note that typically π will be high ($\pi \approx 1$). Let F_X be the cumulative distribution function of X. Our model is then defined by setting the required collateral C to the π-percentile probable capital outflow

[2]Decentralized insurance rests on the idea of a decentralized implementation on a smart contract platform such as Ethereum. See: http://ethereum.org.

due to claims:

$$C(n) = F_X^{-1}(\pi). \tag{1}$$

Let us now to turn to calculating the set of premiums P_i ($i = 1, \dots, n$). For one, we must have that $\sum_{i=1}^{n} P_i = C(n) = F_X^{-1}(\pi)$, but we are free to choose how to distribute the claim costs among policies. While there are multiple ways to distribute the collateralization cost, we choose a method that has naturally desirable properties: (i) the premium P_i should be proportional to the payout P_i^* (intuitively, the higher the payout of a policy, the more the upfront premium cost); (ii) the premium P_i should be proportional to the insurable event probability p_i (intuitively, the lower the probability of the event, the cheaper the premium). To achieve this relationship, set

$$P_i := \frac{p_i P_i^*}{\sum_j p_j P_j^*} F_X^{-1}(\pi)$$

and it is clear that (1) holds. Moreover, it is easy to check that $P_i \to 0$ as $\pi \to 0$ and $P_i \to \infty$ as $P_i^* \to \infty$ as required. If we find that at a small n our premiums are too expensive, we have the option of reducing premiums with some initial subsidy capital seeded into the risk pool; however, we will not pursue the mathematical details of this extension as we believe premiums will already be sufficiently small.

A straightforward calculation shows that the expected value and standard deviation of X are given by

$$E(X) = \sum_{i=1}^{n} p_i P_i^* \ ; \quad \sigma_X = \sqrt{\sum_{i=1}^{n} p_i (1 - p_i)(P_i^*)^2}.$$

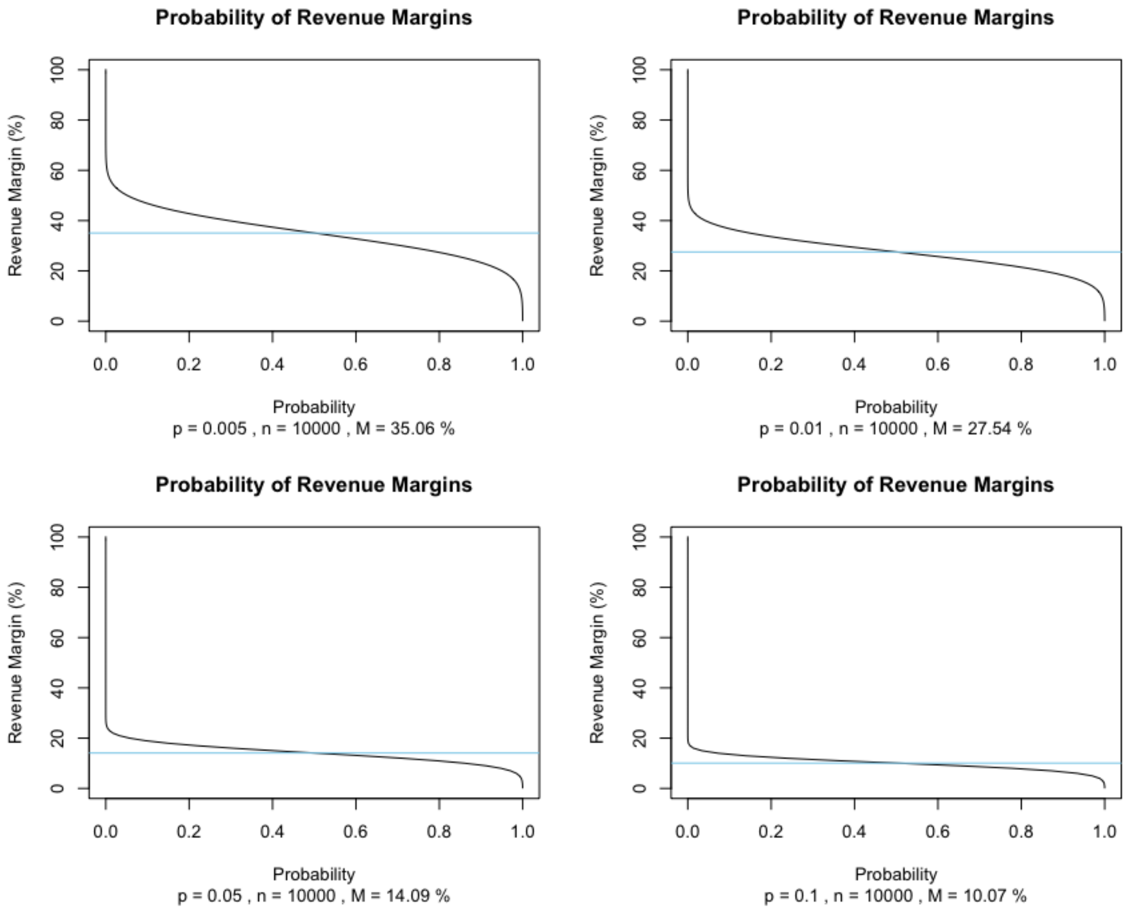


Figure 1: Probability distributions of revenue margins given different values of average event probability p. M is the *revenue margin*, the difference between capital outflows due to claims and the collateral C, divided by C.

We now demonstrate that, given collateralization, our model produces a non-zero expectation of revenue with π-confidence. To show this, let us define the revenue R as the excess capital remaining in the pool after capital outflows due to claims

$$R(n) := C(n) - X(n)$$

and note that $R(n) > 0$ with probability π. (Observe that $E(R) = E(C) - E(X) = F_X^{-1}(\pi) - F_X^{-1}(.50)$ and because of our self imposed constraint that $\pi \approx 1 > .50$ we have $E(C) > E(X)$.) More explicitly, note that the mean and standard deviation of revenue under this model is given by:

$$E(R) = F_X^{-1} - \sum_i p_i P_i^* \ ; \quad \sigma_R = \sqrt{\text{var}(C) + \text{var}(X)} = \sigma_X.$$

We now summarize the inputs and outputs of the Etherisc credit risk model.

Model inputs

1. A vector of insurable event probabilities $\mathbf{p} = \langle p_i \rangle$, $(i = 1, \ldots, n)$.
2. A vector of desired payouts $\mathbf{P}^* = \langle P_i^* \rangle$, $(i = 1, \ldots, n)$.
3. A confidence level for portfolio solvency π, where $.5 < \pi < 1$.

Model outputs

1. **Total Liability.** Total portfolio liability is given by

$$L(n) := \sum_i P_i^*$$

.

2. **Collateral.** Required minimal collateral is given by

$$C(n) := F_X^{-1}$$

.

3. **Excess Liability.** The excess portfolio liability on offer to a reinsurance market is

$$\tilde{L}(n) := L(n) - C(n)$$

.

4. **Premiums.** A vector of premiums corresponding to the payouts $\mathbf{P}^*$, given by

$$\mathbf{P} := \left\langle \frac{p_i P_i^*}{\sum_j p_j P_j^*} F_X^{-1}(\pi) \right\rangle .$$

5. **Capital Outflow.** The expected capital outflow due to claims is $E(X) = \sum_i p_i P_i^*$. The standard deviation of capital outflow due to claims is

$$\sigma_X = \sqrt{\sum_{i=1}^{n} p_i(1 - p_i)(P_i^*)^2}$$

.

6. **Expected Revenue.** The expected revenue for the portfolio is $E(R) = C(n) - \sum_{i=1}^{n} p_i P_i^*$. The standard deviation of revenue is

$$\sigma_R = \sigma_X = \sqrt{\sum_{i=1}^{n} p_i(1 - p_i)(P_i^*)^2}$$

.

Properties We now summarize the general mathematical properties of the credit risk model we have described.

1. The total liability of the risk portfolio increases with the number n of policies, but the required collateralization (that is, $C(n)/L(n)$) tends to decrease with n.
2. Premiums become less expensive as the number of policies grows; that is, $P_i \to 0$ as $n \to \infty$. Premiums are also proportional to the corresponding payout and insurable event probability under the premium calculation we have described.
3. In general, the premiums can be calculated using any algorithm that distributes the required collateral C among n policy holders. A reasonable framework for this calculation is that

$$P_i = kp_iP_i^*$$

   should hold for the i-th policy for some constant $k > 0$.
4. Expected revenue is inversely proportional to the insurable event probabilities and standard deviation of revenue decreases as probabilities decrease. Unsurprisingly, the variance in revenue is the variance of capital outflows due to claims.
5. As demonstrated in Figure 1, revenue margins decrease with higher efficiency of the system (higher n) and lower average event probability p.

8.2.4 Discussion of practical applications

In practice, an insurance risk pool operates continuously: it has some number of valid policies currently being underwritten and creating liability as well as incoming requests for new policies to be underwritten. In a practical context, the pricing of premiums is directly related to marginal changes of the required collateralization C of the risk portfolio.

For instance, suppose the current state of the portfolio requires collateral C, but a new policy is requested against a new or existing event. Recalculation of the model will result in a new required collateralization C' which must be maintained for π-confidence of solvency (and as the portfolio is taking on more risk, $C' > C$). Since the standing premiums in the portfolio have already been remitted, their values cannot be changed. Thus the fair price of a new premium is $\Delta C := C' - C$, guaranteeing that π-confidence is maintained.

In general, the price of a premium should correlate positively with both the payout and the probability of the insurable event: that is, $P_i = kp_iP_i^*$ for some constant $k > 0$. In practice, ΔC may exceed this "expected" baseline premium, especially when the portfolio is small, and so subsidization of the risk pool may be required to bring premiums to expected levels. This scenario is covered in the Etherisc whitepaper through revenue reallocation.

8.2.5 Model estimation

Calculation The core complexity of the model estimation algorithm is the non-triviality of the distribution of random variable X. In order to find C, we must estimate $F_X^{-1}(\pi)$ which is computationally difficult to do directly. Instead, we take an estimation approach by taking some large N, and simulating N random outcomes for the value of X, followed by running a known percentile estimation algorithm on the random outcome for the π-percentile. The rest of the model outputs follow by straightforward calculation based on Section 8.2.3 above.

Python simulation The model described above is implemented as a Python simulation. Please see:

`https://github.com/etherisc/hackathon/tree/master/etherisc-simulator`

Example outputs The following simulation demonstrates a calculation of the insurance model on a set of 60 real flights given actual flight delay estimates from FlightStats. In this calculation, we have set a fixed payout of \$250 for each policy. The model has determined the total liability L of the portfolio to be \$15,000 and a required collateralization $C = \$3,750$, a 25% collateralization of the portfolio. The expected revenue $R = \$2,337.37$ with a standard deviation of \$561.27.

As discussed, note that the theoretical premiums displayed below are different from what would be provided to customers. In general, the premiums can be adjusted lower or higher using subsidization of the risk pool. In practice, premiums will also include additional fixed service fees.

```
# Etherisc insurance calculation


    n:    60
    mu:   1412.63
    sd:   561.27
    L:    $15000.00
    C:    $3750.00
    %:    25.00
    r:    4.00
    R:    $2337.37
```

	prob	premium	payout
DL_762_ATL_MDW	0.018182	12.066488	250
WN_349_RIC_ATL	0.022727	15.083110	250
WN_349_ATL_CMH	0.022727	15.083110	250
WN_203_BNA_SAT	0.026316	17.464685	250
DL_780_ATL_CVG	0.040000	26.546313	250
DL_762_MDW_ATL	0.040000	26.546313	250
KL_724_HAV_AMS	0.040816	27.088050	250
DL_132_TPA_DTW	0.046512	30.867805	250
SK_904_EWR_ARN	0.048387	32.112468	250
DL_160_MSP_AMS	0.048387	32.112468	250
DL_132_DTW_AMS	0.052632	34.929329	250
WN_203_MDW_BNA	0.052632	34.929329	250
AC_36_BNE_YVR	0.053571	35.553061	250
KL_642_JFK_AMS	0.064516	42.816613	250
DL_160_IND_MSP	0.064516	42.816613	250
DL_1527_ATL_FLL	0.065574	43.518511	250
DL_476_JFK_BCN	0.066667	44.243829	250
WN_349_CMH_MCO	0.068182	45.249369	250
AA_1033_DFW_RSW	0.073171	48.560297	250
DL_2452_ATL_RIC	0.075472	50.087360	250
WN_203_ABQ_BWI	0.081081	53.810073	250
DL_337_ATL_NAS	0.083333	55.304776	250
DL_1527_FLL_ATL	0.084746	56.242159	250
DL_142_LAS_SEA	0.093023	61.735571	250
AA_66_SFO_JFK	0.096774	64.224937	250
SK_903_ARN_EWR	0.096774	64.224937	250
LA_3010_BOG_MDE	0.098361	65.277789	250
DL_2452_RIC_ATL	0.098361	65.277789	250
DL_72_MCO_ATL	0.098361	65.277789	250
YV_6273_IAH_ELP	0.100000	66.365763	250
DL_54_ATL_LOS	0.100000	66.365763	250
EV_5230_ATL_BTR	0.111111	73.739715	250
F9_1539_ATL_PHX	0.111111	73.739715	250
WN_258_GSP_ATL	0.111111	73.739715	250
WN_258_PHL_TPA	0.111111	73.739715	250
LA_800_AKL_SCL	0.112903	74.929082	250
DL_72_ATL_AMS	0.112903	74.929082	250

NZ_29_IAH_AKL	0.113636	75.415629	250
EV_5597_LFT_ATL	0.114754	76.157406	250
EV_5597_ATL_LFT	0.114754	76.157406	250
KL_624_ATL_AMS	0.117647	78.077347	250
EV_5230_ATL_FAY	0.117647	78.077347	250
EV_5230_FAY_ATL	0.117647	78.077347	250
KL_678_YYC_AMS	0.118644	78.739020	250
OO_4568_SLC_PHX	0.120000	79.638900	250
DL_907_DTW_RDU	0.120000	79.638900	250
DL_54_IAD_ATL	0.128205	85.084289	250
LA_800_SYD_AKL	0.129032	85.633220	250
AA_83_JFK_LAX	0.129032	85.633220	250
KL_652_IAD_AMS	0.129032	85.633220	250
KL_606_SFO_AMS	0.129032	85.633220	250
QR_755_DOH_ATL	0.131148	87.037061	250
WN_203_BWI_MDW	0.131579	87.323337	250
DL_477_BCN_JFK	0.133333	88.487664	250
HA_444_BNE_HNL	0.137931	91.538975	250
DL_476_LAX_JFK	0.142857	94.808205	250
DL_675_NAS_ATL	0.145161	96.337365	250
LA_3508_BOG_CUN	0.145161	96.337365	250
JJ_8000_GRU_BOG	0.145161	96.337365	250
NZ_10_AKL_HNL	0.147059	97.596702	250

Figure 2: Example calculation on a 60-policy insurance portfolio using actual flight delay data.