Trading Service Level Agreements within a Peer-to-Peer Market

Ioan Petri\textsuperscript{1} Omer Rana\textsuperscript{2} Gheorghe Cosmin Silaghi\textsuperscript{1}

Business Information Systems, Babes-Bolyai University, Romania

School of Computer Science & Informatics, Cardiff University, UK

26.10.2010
Table of contents

1. The concept of complementary currency
2. Related work
3. Protocol stages
4. SLA significations
5. Results
6. Conclusions
7. Future Work & Usage
The concept of complementary currency

Related work

Protocol stages

SLA significations

Results

Conclusions

Future Work & Usage

Provider: $p - RP$

Client: $EV - p$

For a group of providers

“Welfare maximization”

Results + penalty

Quality of Service Attributes

SLA: \{(SLO, SLI, BV, pen)\}

SLI: values (int/float), constraints (numeric/range, logic)

BV: value (int)

function ($f(\text{provider capability})$)

Expected Value (EV) (Client)

Penalty/Reward

Reservation Price (RP) (Provider)

View SLA as a virtual currency in an “options” market

Ioan Petri, Omer Rana, Gheorghe Cosmin Silaghi

Trading Service Level Agreements within a Peer-to-Peer Market
General idea

Using computing resources as tradable objects in P2P systems – with SLA based access to these objects.

A key observation

An SLA is itself a “resource” – may be traded as an “option”. As it refers to an allocation that has yet to take place.

- **User model**: Availability of capacity that is not of interest to a user directly – but something already payed for.
- **Provider model**: Ability to support capacity planning – once sold, provider agrees to offer service. No need to re-establish contract or look for alternative buyers.
- **Market model**: Ability to aggregate capacity from multiple providers to provide a combined service.

Focus: How can we vary demand for SLAs, identify how SLAs are traded, and price SLAs to provide “welfare” within a community of users and providers.
Complementary Currencies

Identify commonality with the use of complementary currencies – especially the circulation of such a currency.

- Work alongside a real currency – and need to be mappable – and also called “local money”
- Key issue: rate of exchange, circulation scope & value fluctuation over time
- Some currencies backed by real assets – e.g. gold, oil – and more interestingly services (i.e. amount of time spent by individual A for activity B)

Example usage:

- Encourages *localisation* of trade & relationships
- Encourages use of under-used resources
- Recognises an informal economy
We use SLAs (Service Level Agreements) as tradeable objects within a P2P system

- SLAs are particularly useful when considering interactions in environments with limited trust between participants
- An SLA may be used for: (i) an expression/proof of debts as well as credits –debts to the client and credits to the service provider; (ii) as a token of exchange between participants; (iii) as an identification of responsibilities of participants involved (such as the client and service provider)
- An SLA refers to a service that is to be provided in the future, thus it is possible to consider an SLA-based options market
Previous alternative systems

- **WAT** is a debt-oriented (i.e. a capability is used before payment) approach based on WAT tickets, and provides an alternative currency to facilitate local exchanges.

- **iWAT** is the internet version of WAT that implements the WAT core and examines the problem of ticket exchange by designating the drawer as the responsible party for ticket authenticity.

- **Samsara** concentrates on establishing service relationships between peers, minimising risk due to failure and claim, to ensure equivalence between the contribution and consumption of peer nodes.

- **PPay** is a token-based system which implements a protocol for coin assignment at the level of peer nodes using signature generation, verification and network messages.
Figure: SLA Use: Time Line

- $\Delta T_0 = T_1 - T_0$: SLA held by the initial client
- $T_1$: Client node no longer needs service identified in the SLA; or Client gets an offer from another node. Client forwards SLA and receives payment.
- $\Delta T_1$: forwarding (and payment) process may continue a number of times.
- $\Delta T_2$ represents the time over which service execution takes place, and at $T_3$ the SLA expires and is no longer valid.
The concept of complementary currency
Related work
Protocol stages
SLA significations
Results
Conclusions
Future Work & Usage

The workflow

Ioan Petri, Omer Rana, Gheorghe Cosmin Silaghi
Trading Service Level Agreements within a Peer-to-Peer Market
The details of protocol

Operations

- SLA issuing $t_0 - t_1$
- SLA forwarding $t_1 - t_2, t_2 - t_3$
- SLA redemption $t_3$

Two types of provision:

- direct provision – those that it provisions through it’s locally owned resources (corresponds to SLA creation)
- indirect provision – those that it defers to other providers through previously acquired SLAs (involves advertising SLAs that a provider already holds with others)
The details of protocol

Roles and responsibilities

- For issuing – \([T(B), \{t_1, t_2, ..., t_n\}]\) identifies the list of services that node B can provide, where \(t_i = SLA^X_B\) is an SLA that node B has with node X, and which node B is willing to trade. At this point node A holds an SLA with B \((SLA^B_A)\) which has a monetary value (equivalent to what node A has paid node B).

- For forwarding – \([T(A), \{t_1, t_2, ..., t_m\}]\) is the list of service that node A can provide. The client of the agreement has changed, as \(SLA^B_A\) becomes \(SLA^B_C\), and the value of \(SLA^B_A\) is different from \(SLA^B_C\).

- For redemption – occurs after an SLA has been agreed between two parties, and may take place after several forwarding operations. In the previous example, node D redeems the SLA with provider B.
We calculate the economic benefit expressed in terms of profit of loss.

- Cost during exchanges: $C_s = \sum_{i=1}^{k} P_{SLA_i}$; where $P_{SLA}$ represents the price of one acquired SLA; $k$ represents the number of SLAs that are acquired during the experiment;

- Income during exchanges: $I_s = \sum_{i=1}^{k} P_{SLA_i}$; where $P_{SLA}$ represents the price of a sold SLA; $k$ represents the number of SLAs that are sold during the experiment.

According to a probability distribution the system can express statuses by: Profit: $P_s = I_s - C_s$; and Loss: $L_s = C_s - I_s$. 
Simulation environment

Simulator

PeerSim: Open source, Java-based simulation framework for developing and testing P2P algorithms in a dynamic environment.

Our simulation:

- A *pay-before-use* payment scheme. Once paid, actual currency is not involved.
- Peer $p_i$ provides one or more services $s_i$ of type $t_i$. One type per peer in simulation.
- Type: storage, computation or aggregate – can be user defined
- $P = \{p_1, p_2, p_3, \ldots, p_n\}$ – list of types $T = \{t_1, t_2, t_3, \ldots, t_m\}$, where $t_i$ represents the type of service that a set of peers $P_j = \{p_1, p_2, p_3, \ldots, p_k\}$, $(P_j \subset P)$
- Demand based on a *view* parameter – how many peers are requesting a particular type of service at a given point in the simulation
Simulation Objectives

- SLA value fluctuates based on level of demand in the system – configuration parameter – defined by $\text{view}$

$$V_{type_{c_i+1}} = V_{type_{c_i}} \times f(d)$$

- where $c_i$ represents the $i^{th}$ cycle of the simulation

- Evaluate the variation of budget (at start and at $i^{th}$ cycle) associated with a particular peer, budget distribution: $B_i = \{b_1, b_2, b_3, ..., b_n\}$.

- Identify a “Welfare” ($W$) metric – indicating the value benefit observed by a Peer over time – $W = \sum_{i=1}^{n} b_i$
Demand is varied from 20 peers to 10 peers in view for each $t_i$, $i = 1, 10$. View for each peer type set to 20 requesting peers.

Configuration with 10 maximum issuing nodes and 10 maximum SLAs per node.

After 60 cycles – no further SLAs being exchanged in the system – therefore no further trading takes place.
Experiment 2

- Demand uses 10 peers in view for each $t_i$ type (for $i = 2, 10$) and the view for $t_1$ being changed from 20 to 15 peers. 10 maximum issuing nodes and 10 maximum SLAs per node.

- Welfare affected by changes in any particular type – and relationship between types
10 maximum issuing nodes and 10 max. SLAs per node – demonstrates Welfare with different SLA types.

Demand is configured at 10 requesting peers for each $t_i, i = 1, 10$, in the context of 5, respectively 10 SLA types.

Impact on Welfare based on diversity of SLA types being exchanged – keeping demand constant.
Execution Probability: controls transition between the issuing and the forwarding process.

The experiment is configured to use 10 maximum issuing nodes and 10 maximum SLAs per node. The demand is configured to 20 requesting peers for each $t_i$ type, $i = 1, 10$.

Increase in exchange increases welfare
Use 10 SLA types, with significant demand for only one type, in this case $t_1$, and less demand for other types $t_2-t_4$.

Keep user count fixed – hence increase in one type influences other types.

Relevant when considering resource aggregates – i.e. combining storage and computation.
The benefit of the system is influenced by:

- the number of participants
- the level of demand – identified by the “view” parameter in our simulation
- type of SLAs used – and variation in demand based on SLA types

Using the view of peers (i.e. how many partners a peer can trade with) we demonstrate that:

- the benefit depends on the view parameter
- greater exchange leads to greater welfare – and influenced by the time period over which exchange takes place (execution probability)
Scalability:
- Increasing number of participants in the market.
- Increase diversity of SLA and participant types.

Real business applicability:
- Use within a Re-seller market
- Used by a broker to support service aggregation