A Mechanism for Charging System Self-Configuration in Next Generation Mobile Networks

Ralph Kühne[1,2], Gerald Görmer[1], Morten Schläger[1], Georg Carle[2]
[1] Siemens Networks, Siemensdamm 62, 13627 Berlin, Germany
[2] University of Tübingen, Sand 13, 72076 Tübingen, Germany

***Abstract*-- Self-management is an emerging idea in charging system research taken over from autonomic computing to address the complexity problem of system deployment, maintenance and administration and to improve the performance and flexibility of the charging system what will as a whole eventually decrease its total cost of ownership. In this paper, we describe a mechanism for charging system self-configuration when new network or service elements that record charging information are added as well as for charging system self-optimisation during every-day operation.**

***Index terms*-- charging, self-configuration, self-optimisation, next generation mobile networks**

INTRODUCTION AND MOTIVATION

In this conceptual paper we introduce a mechanism for enabling self-configurability and self-optimisation in charging systems for use in next generation mobile networks. The self-management idea is taken over from autonomic computing [1,2] in order to address the complexity problem of system deployment, maintenance and administration and to improve the performance and flexibility of the charging system. Our approach builds on the logical charging architecture as devised by the 3GPP (Third Generation Partnership Project) in Release 6 [3] and comprises an overall solution for the flexible and efficient configuration of the so-called Charging Trigger Functions (CTFs), i.e. specialised accounting components on the relevant network and service elements. In today's mobile telecommunication networks such an overall solution does not yet exist and the coordination and synchronisation problem, which is caused by technological diversity and the distributed and mostly independent nature of the charging processes carried out by the different CTFs on the different charging levels (bearer, subsystem and service) of the 3GPP architecture, is not addressed.

Although a lot of research work has already been done regarding policy-based solutions for accounting and charging (e.g. [4], [5], [6], [7]), these concepts still lack the self-management features. However, such features bear the potential to significantly and lastingly reduce the deployment cost and the operational expenses, i.e. eventually the total cost of ownership, by a more autonomic mode of operation in order to realise the desired efficiency and flexibility.

CHARGING SYSTEM SELF-MANAGEMENT ASPECTS

When looking at the processes and procedures that are relevant during the life cycle of a charging system or its components, several aspects become apparent, to which the idea of self-management especially with respect to self-configuration and self-optimisation can be applied. On closer examination, a classification of these aspects seems possible according to both the focus of self-management and the impact of its processes.

Regarding the focus, self-configuration and self-optimisation may on the one hand address the level of pricing and tariffing (*economic focus*) and on the other hand, the level of system configuration regarding charging and accounting processes (*technical focus*). Regarding their impact, the self-management efforts of the charging system may take effect in a direct feedback loop affecting the currently running service session or transaction (*direct feedback*) or be mainly based on historical performance information or billing data. The latter case will be characterised by taking place offline and only affecting later sessions or transactions (*indirect feedback*).

The result of combining these two possibilities of classifying self-management aspects is shown in Table 1 and further elaborated on in the following subsections.

Technical and Economic Focus

For the technical focus, self-management refers to the charging system's operational parameterisation (e.g. reporting intervals, etc.) as well as its structure, composition and interaction (e.g. participating charging-relevant entities, synchronisation of charging processes) and aims at increasing system performance. A concrete objective here may be to collect exactly that minimal information set that is needed to carry out the charging task successfully, i.e. satisfying the information needs of the operator and the customer to settle the claims.

In contrast, the economic focus refers to the tariffs that an operator wants to apply to the services he offers as well as the overall tariff models he employs. Here, not the charging sys-

TABLE I. CHARGING SELF-MANAGEMENT ASPECTS AND EXAMPLES.

		Focus	
		technical (operational parameterisation, structure, composition and interaction)	**economic** (prices and tariffs/tariff models)
Feed-back	**direct** (online)	*change of reporting intervals, adaptation of assigned quota size, change of measurement point/addition of measurement point for single active service session*	*dynamic price re-negotiation/price adaptation when system load changes/during congestion*
	indirect (offline)	*general change of employed reporting intervals/assigned standard quota size, change of pre-defined measurement point/addition of measurement point, deployment of new services and integration of service elements into the system; for one customer/customer group/product offer or system-wide*	*analysis and adaptation of tariff models from a profit maximisation's point of view*

tem itself is re-configured or optimised in its structure or interaction but tariff or pricing aspects. Objectives therefore have an economic character, such as profit-maximisation or influencing customer behaviour and may be manifold depending on the respective operator. Changes here affect what information is needed from the charging system (e.g. to collect volume information) and what action (e.g. rating, direct debiting of a pre-paid account) shall be taken by the charging system in order to bill the respective customer correctly.

The operator's choice is nevertheless limited by the technical capabilities of the charging system. This fact already demonstrates the mutual inter-relationship between the two self-management foci. In addition, as the applied tariff models govern the information needs, for instance by defining the charging model to be used (e.g. time-, volume-) and how the collected usage information shall be treated (online/offline charging), tariff changes will often have a direct impact on the configuration of the charging system from a technical point of view.

Direct and Indirect Feedback

If the self-management effort shall effect the currently on-going service session or transaction, it has to take place during the respective session or transaction (online). Therefore, it comprises a *direct feedback loop* immediately influencing the charging process of a currently on-going service usage. Additionally, there are accompanying hard real-time requirements regarding their completion in order for their optimising adaptations to be of use. As the resource consumption progresses, additional participants join or the environment changes (e.g. network load increases caused by other customers), these adaptations may either try to improve the system configuration (e.g. adapting the quota size or the reporting interval) or the pricing (e.g. offering a lower price for decreased bandwidth, QoS, etc.). Depending on the optimised parameter, the adaptations may be transient and purely operational in nature, i.e. they may not necessarily influence later service usages.

Besides, self-optimisation may also address objects beyond that of a single on-going session or transaction, like all sessions belonging to one service, to a customer or to customer groups or even affect the entire system. In this case, the self-optimisation is carried out after the respective service sessions or transactions have completed which provide the information on whose basis this self-optimisation takes place (offline). Therefore, such offline self-optimisation processes comprise an *indirect feedback loop* effecting later sessions or transactions and also lead to changes of the system configuration and of tariff aspects that are much broader in scope. In contrast to a direct feedback, where the service usage causes changes in its own charging, indirect feedback denotes a self-management process that builds on historical data of potentially any kind and takes effect on future service usages.

INITIAL SELF-CONFIGURATION OF CHARGING SYSTEMS AND THEIR COMPONENTS

After having given an overview of possible self-management aspects for charging systems, we describe in this section a concrete mechanism for charging self-configuration directed at technical online and offline optimisation enabling a general bootstrapping and deployment of CTFs on new charging-relevant service or network elements. The economic focus is not in the scope of the optimisation processes described here.

General Concept and Principles

In general, our approach lets the CTF request additional configuration when charging rules in the initial charging configuration installed during deployment tell it to do so. Furthermore, these charging rules will be used to decide whether to remain passive or become active and in the latter case, what this activity shall be in detail, e.g. requesting quota, creating Charging Data Records (CDRs) and which information to include therein, respectively. The novel functional element responsible for rule generation, modification and provision in our concept is the so-called Charging Configuration Manager or in short *Configuration Manager*.

The approach bears a certain resemblance with the Flow-based Charging (FBC) concept [8] but is much broader in scope as it addresses all CTFs on all charging levels. Besides, FBC's Charging Rules Function (CRF) becomes an integral part of the self-managing charging system, as such charging rules are the primary means of a policy-based charging system self-configuration and -optimisation.

In order for such a policy-based charging system to work in a self-managing way in a self-deployment scenario, several requirements have to be fulfilled:

- the ability of newly added CTFs to find their charging system, or more precisely, their configuration manager
- the existence of a well-defined language to express offered services and charging-relevant capabilities (which events and activities can be recognised, what information can measured and reported and how can reporting take place, e.g. participation in an online charging dialog)
- the configuration manager must be able to interpret these capabilities and offer options for human product offer generation and pricing decisions/tariff selection
- based on the created product offers and the selected tariffs, the configuration manager must be able to create a charging configuration that governs the charging process on the respective CTF (i.e. customised definition of chargeable events to be reported for the charging mechanisms as well as information to contain/transfer in charging events and which elements to contact)
- the CTFs must be able to implement the received initial charging rules/configurations and act accordingly.
- if new services are added and the newly added service elements do not comprise an integrated CTF, they nevertheless have to be able to provide information how the usage of this particular service can be identified (e.g. presence of specific SIP-URI in INVITE request, Packet Data Protocol (PDP) context activation request with specific Application Point Name (APN), etc.). Such elements then take the role of the Application Function (AF) known from FBC [8].

We call a CTF fulfilling the first five of these requirements a *Next Generation Charging Trigger Function* (ngCTF). This is not the case for CTFs in today's 3GPP networks.

IMS Example

In order to make the initial self-configuration feature more tangible, we exemplify its workings for a deployment scenario of a new IMS Application Server (AS) providing some end-user services. The interactions between the involved functional elements are depicted in Figure 1.

This IMS AS is assumed to comprise an integrated ngCTF, which the application(s) running on the server inform of all relevant events and activities taking place in the IMS AS. These events and activities make up the set of chargeable events that the ngCTF may report to the charging system. They are described in a data structure that we call the *charging capabilities description.* Besides the chargeable events the ngCTF is able to recognise and report, this data structure also comprises the charging mechanisms (online, offline) and corresponding protocols (e.g. Diameter Credit Control Application) the ngCTF supports as well as which information can be reported in detail for the different chargeable events and end-user sessions (e.g. time, volume).

If the ngCTF is added to the operator network it either has the configuration manager's address pre-configured or it may depend on a look-up service to obtain this address as depicted in Figure 1. If the address has been obtained, the ngCTF contacts the configuration manager by sending its identifier, its address and its service-related charging capabilities in a regis-

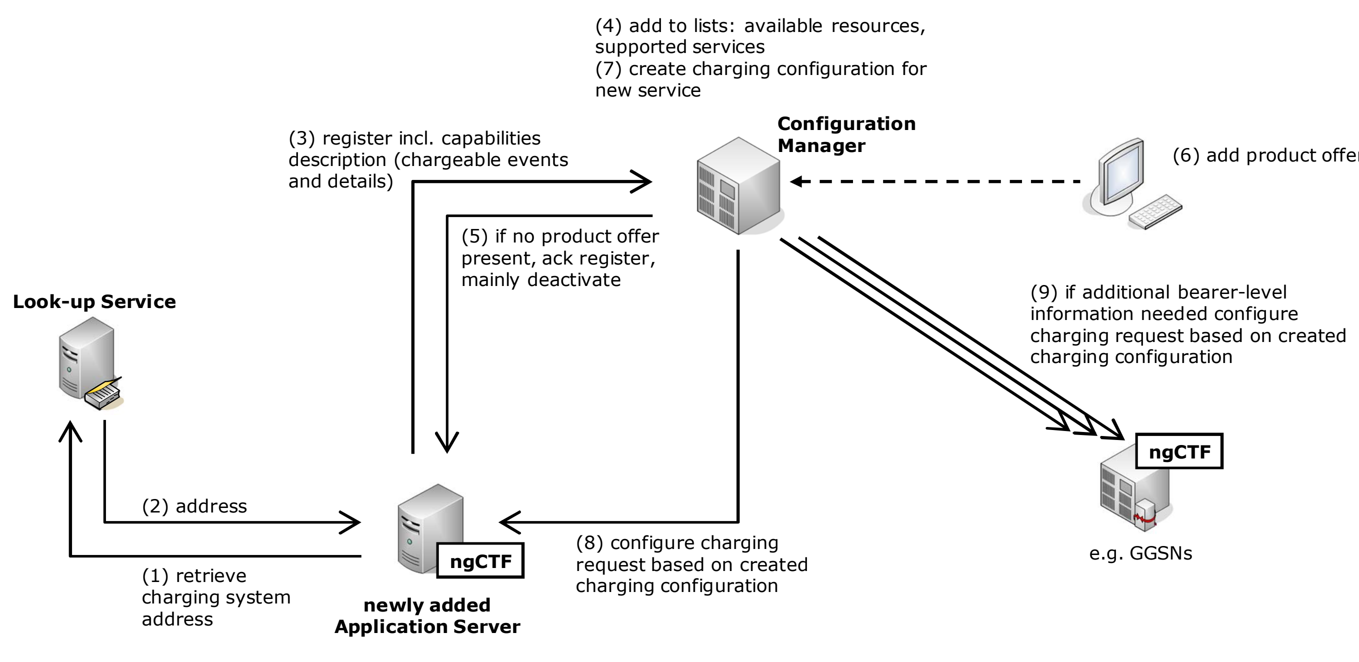

Figure 2. Initial Self-Configuration Example

ter message. This register message may additionally include descriptions of all services the corresponding IMS AS provides. Alternatively, service descriptions may also be provided to the configuration manager in some other way, e.g. by manual input.

Upon reception of the register message, the configuration manager updates two of its data structures. Firstly, there is the *available resources list* to which it adds the ngCTF's identifier and address as well as the ngCTF's charging capabilities. Secondly, the configuration manager adds the services to the *supported services list* for which charging support is indicated by reportable chargeable events in the ngCTF's charging capabilities. This also includes a reference to the providing ngCTF and the service's commonly known identifier and is derived from the above-mentioned service descriptions that define by which chargeable events the respective service's usage can be identified.

Depending on the existence of product offers that already include the services provided by the newly added IMS AS, the consecutive process steps differ. For now, we restrict our considerations to the case where there is no product offer that already contains a service provided by the newly added IMS AS. By *product offer* we mean a single service or a combination of several services that is offered as a whole by the operator. Each product offer has a price associated with it, which is determined by the tariff applicable to the respective subscriber and which, for instance, may depend on time of day, user category, user consumption, etc.

Since the service is not yet included in any product offer, the configuration manager only carries out a basic configuration of the ngCTF by sending a register acknowledgement, which mainly leaves the ngCTF deactivated.

However, if the operator afterwards creates a new product offer that contains one of the newly added services, the configuration manager has to generate a corresponding *charging configuration* for the new ngCTF. Furthermore, if the product offer requires resource consumption details, which the ngCTF in the IMS AS cannot itself provide, other ngCTFs need to be included in the respective charging task. As it is not a priori known which other network elements will be involved in rendering a service, i.e. which ngCTF will have access to the required charging information (e.g. by lying on the data or signalling path), the configuration manager has to add the respective charging configuration for this new service to all potentially involved ngCTFs with the respective capabilities. These ngCTFs are known to the configuration manager by its available resources list. For instance, if the Radio Access Technology (RAT) type is drawn on to derive the price for a service, a service-level ngCTF may not have this information available, such that bearer-level ngCTFs may need to be employed. For example, the Serving GPRS Support Node (SGSN) could be configured to report the RAT type when the PDP context (or a service data flow) is established as well as when inter-system handovers occur.

The sum of all individual configurations relating to the same product offer on different ngCTFs we call the *charging strategy* for this product offer. The optimisation problem that the configuration manager has to solve is therefore twofold. Firstly, it is directed at all charging configurations in a charging strategy for a single product offer and secondly, as the strategies for the different product offers may influence each other's performance, this, too, has to be taken into account when carrying out the optimisation process and deriving the overall (global) charging strategy of the configuration manager.

The rules contained in the charging configuration will not always be completely known beforehand but will contain variables that are parameterised when the service is requested (dynamic rules). In such cases, the ngCTF contacts the configuration manager (taking over the role of a CRF known from FBC) to request the missing parts or even entire rules. Missing parts may for instance be information about how the service usage may be identified. For a packet-bearer ngCTF this could be an IP 5-tuple or a flow label relating to service data flows and for a subsystem- or service-level ngCTF a SIP Uniform Resource Identifier (URI), SDP (Service Description Protocol) fields in the SIP message's body or other service-specific primitives.

As already indicated, the self-deployment of a bearer- or subsystem-level element that in contrast to a service-level element does not directly provide an end-user service, like a SGSN or an Serving-Call State Control Function (S-CSCF), includes the provision of configurations relating to all services present in product offers relevant to this respective element. Rules in such configurations are likely to comprise a larger number of dynamic rules than with service-level elements and a re-configuration will become necessary more often, namely whenever relevant services are added or their bundling in product offers are changed. Despite these facts, the process of self-deployment as described in the example above will be largely identical.

Permanent Self-Optimisation During Every-Day Operation

The deployment phase has provided the ngCTF with an initial charging configuration, which comprises rules for all services contained in product offers and being of relevance to the ngCTF. This means that after the initial self-configuration of the newly added ngCTF, it is able to support the offline and online charging processes by providing the needed charging information to charge and bill the resource consuming users correctly and efficiently.

Concept and Principles

It is the responsibility of the configuration manager to maintain this optimal state by making efficient usage of the charging-relevant resources at its disposal, i.e. for the scope of this article, all assigned ngCTFs. Especially, if during every-day operation the initial or also prior configuration decisions that at their time seemed satisfying may turn out to be inefficient or insufficient to successfully complete the respective charging task, countermeasures have to be taken to remedy these unwanted situations. The configuration manager's means to this end are again the charging configurations that control the ngCTFs behaviour and that make up the charging strategies for the different product offers.

Reasons for such developments leading to inefficient or insufficient results may be:

- changes in the charging system environment, such as shifts in customer usage patterns, changes in service usage behaviour, roaming behaviour, etc.
- unexpected problems, like ngCTF failure or the missing of charging details in reports to the configuration manager
- operator-induced changes (e.g. of tariffs, product offers, or temporary special offers) that demand for a system reconfiguration at least of those parts of the system related to the affected services/product offers.

Performance reports and health signals from the ngCTFs form the basis for the configuration manager's recognising unfavourable developments stemming from failures or changes in the system environment as they cannot be directly communicated to the configuration manager as operator-induced changes would.

In order to decide about the effectiveness of a charging configuration and the sufficiency of its current realisation, the following requirements have to be fulfilled:

- performance metrics have to be identified that allow for a performance evaluation by the configuration manager,
- ngCTFs must have self-monitoring capabilities to report their performance to the configuration manager using the identified performance metrics in form of *performance data* as well as their "system health" in form of *health signals*,
- the configuration manager must be capable of a performance and "health" evaluation based on the ngCTF-provided performance data and health signals,
- the configuration manager must know how the corresponding performance parameters can be influenced towards a potentially unknown or varying optimum, and
- the configuration manager must know ways how "unhealthy" states can be remedied as well as have the necessary means to do so,
- for operator-induced re-configurations, the configuration manager must have appropriate means to derive a new initial configuration for the affected ngCTFs, a new charging strategy as well as an overall strategy.

Depending on the exact cause, the self-management process is either self-optimising or self-healing in nature. The first mainly applies to the inefficiency of a configuration, i.e. the charging task or tasks are carried out correctly, but the resources are not optimally employed. In contrast, the latter relates to the insufficiency of the configuration's realisation, i.e. the charging task or tasks cannot be completely carried out and the information need for charging and billing the subscriber cannot be satisfied.

If the configuration manager recognises either one of the two cases by means of unsatisfying performance data or bad health signals received from the assigned ngCTFs, the self-adaptation process is initiated resulting in the execution of an autonomic feedback loop (see e.g. [2, 1]) as shown in Figure 2. Based on the analysis of the received information (if the analysis includes early indicators of unfavourable developments, the taken measures could also be proactive), a plan of countermeasures is derived, i.e. in the context at hand, new charging configurations are generated. Potential steps to validate this plan regarding possible side effects that may counter-effect its desired results or cause problems or inefficiencies in other system parts may be carried out before the plan is executed, i.e. it has to be verified that the new charging strategy does not negatively influence the performance of other charging strategies. Execution here means that the old configurations on the affected ngCTF are replaced by the new ones. The extent of this replacement depends on the triggering problem and the resulting new configurations to solve it.

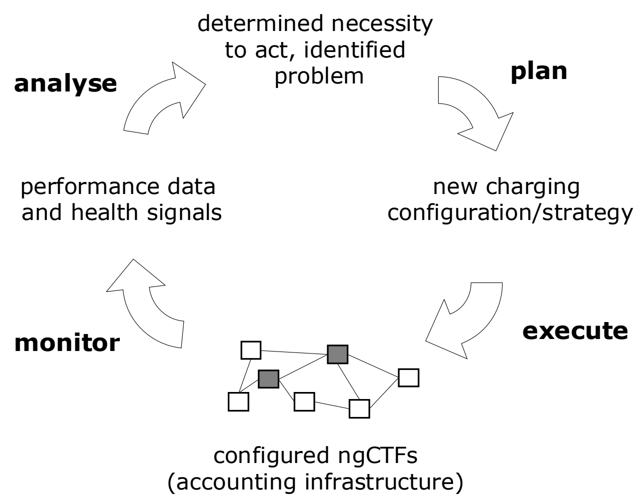

Figure 2. Self-Adaptation Feedback Loop

Examples

In this section, we give two examples for self-optimisation and self-healing in the charging context.

The first assumes unexpected user interest or other incomplete information at the time of charging configuration generation, that lead to an intolerable load in the respective ngCTFs. In such a case, the configuration manager would have to adapt this charging configuration by finding other possible ngCTFs that are also able to provide the necessary information without the mentioned drawback. If for example a product offer refers to a content service where a certain price has to be paid for the content (e.g. MP3 files, ring tones, etc.) and additionally a volume-based transport fee is levied on transferring this content, the respective bearer-level ngCTF (e.g. GGSN) could be relieved by having the content server also report the size of the transferred files to derive the overall charge. After analysis has lead to this conclusion, the configuration manager adapts all affected ngCTFs' configurations and disseminates these changes to them for future use (technical offline optimisation). Based on subsequent performance data or health signals, the configuration manager controls the success of the re-configuration. Optionally, the configuration manager may also try to predict the effect of the new configuration by means of internal models or heuristics as well as carry out a potential conflict resolution that might otherwise lead to sub-optimal results.

Another example for charging system self-adaptation are cases where the operator introduces new tariff models, creates new product offers or changes existing ones. As a result, the current charging configuration of the assigned ngCTFs may not match the accompanying information need anymore (insufficiency of charging configuration) or generate charging information that is superfluous (inefficiency of charging configuration). In both cases, the configuration manager has to generate new configurations for all affected ngCTFs. For instance, if the operator switches from a stepwise pricing of multimedia messages (e.g. every 250 kilobytes the price of one message is added to the overall price) to a per-piece pricing where the number of messages decides about the charge and not the size of each message, a configuration that lets a bearer-level ngCTF measure the size of multimedia messages would be inefficient. The configuration manager should therefore deactivate volume-based accounting on the bearer-level ngCTFs and instruct the MMS Relay/Server to report the number of multimedia messages sent by users.

CONCLUSION AND OUTLOOK

As we described in the previous sections, system self-management strives at decreasing the total cost of ownership of IT systems, i.e. in the context of this paper, of charging systems. We have presented mechanisms for charging system self-configuration when new ngCTFs are added as well as described concepts for self-optimisation during its every-day operation.

Besides the technical focus, on which we concentrated in this paper, an additional economic self-optimisation extends the cost reduction objective by also aiming at increasing the overall systems profitability of providing services to customers. This is achieved by adapting prices and tariffs to reflect the current load situation or to better address the customers' individual needs. For example, the configuration manager could evaluate the tariff models from a profit maximisation's point of view. Here, alternative tariffs are examined and evaluated on the basis of recorded service usage data as well as customer models that try to predict customer behaviour when tariffs are changed. Both aspects are implemented by offline functionality that interfaces with the respective operator databases containing the necessary information, such as system design and available infrastructure components or service usage information. The inner workings are mainly simulation based and results are suggested to the operator in form of changes to existing tariffs or as new tariffs. The operator then decides how to proceed.

Adding the economic aspect is in a way more tempting than just concentrating on technical optimisations, as it offers reduction in costs and a potential increase in income at the same time. On the other hand, the actual results from changes of prices and tariffs may eventually proof difficult or impossible to predict because of unexpected or irrational customer or market reaction. The higher potential benefits are therefore confronted with a higher complexity of the solution which includes the demand for realistic models of the environment.

Benefits

To summarise, expected benefits for charging system self-management in its different facets are:

1. automatic and therefore faster and less costly integration of new measurement points into an existing charging infrastructure, i.e. decrease in deployment and customisation costs (especially with online charging)
2. automatic re-configuration of measure points already in operation, e.g. because of tariff changes, i.e. decrease in administration costs
3. faster and less costly deployment of new services without the need of extensive manual re-configurations beyond the actual scope of the service (c.f. bearer modifications for MMS charging) to achieve correct charging of this service's usage, i.e. shorter time-to-market with less deployment costs
4. higher flexibility to quickly adapt to market trends by easy possibility to change tariffs, especially regarding employed charging models
5. higher system efficiency by charging-system self-optimisation thereby with given system higher load possible (support of more customers per system unit), i.e. higher return on investment
6. increased profitability of the operator network by adapting tariffs and prices to optimise revenue and influence customer behaviour

Potential Next Steps

As described in this paper, the charging system is designed to be intelligent, controlling and configuring the ngCTFs on the accounting layer. These ngCTFs are configurable and also return information about their status and performance but they do not themselves take autonomic action. Rather, the charging system does, based on the information the ngCTFs deliver. This is the first step, which we have described in this paper.

In a next step, the accounting layer could be made intelligent as well, such that the ngCTFs implement a second but restricted layer of optimisation. ngCTFs will try to find a local optimum for their performance but these individual efforts will be orchestrated by the intelligent charging system that knows the overall (global) state of the system and its performance. Decentralised self-optimisation processes would be carried out based on self-monitoring data as well as health and status signals from adjacent CTFs, but under the centralised charging systems coordination.

As current considerations only distribute the accounting task on the different CTFs, the same could be done for the charging part, namely mainly the rating process. Here, the decision could be made to transfer the rating process and the customer account to a trustworthy ngCTF in the network that plays the role of the Online Charging System (OCS) for one or some users. The resulting distributed online charging would offer load balancing capabilities and take the credit pooling concept a step further. But it is for further research, if the benefits of such an approach (mainly higher number of customer supported, decreased requirements on coordinating OCS) outweigh the drawbacks, such as the need for additional functional capabilities in the ngCTFs as well as reliability and transactional requirements resulting in higher investments and operational expenses for such advanced ngCTF.

REFRENCES

[1] Jeffrey O. Kephart, David M. Chess: "The Vision of Autonomic Computing", in: *IEEE Computer*, January 2003, pp. 41 – 50

[2] Michael G. Hinchey, Roy Sterritt: "Self-Managing Software", in: *IEEE Computer*, February 2006, pp. 107 – 108

[3] 3GPP TS 32.240: "Telecommunication management; Charging management; Charging architecture and principles", Release 6

[4] J. Jähnert et al., "The 'pure-IP'Moby Dick 4G architecture", in: *Computer Communications*, Vol. 28 (2005), p. 1014 – 1027

[5] M. Koutsopoulou, A. Kaloxylos and N. Alonistioti: "Charging, Accounting and Billing as a Sophisticated and Reconfigurable Discrete Service for next Generation Mobile Networks", IEEE Vehicular Technology Conference (VTC) Fall 2002, 2002

[6] G. Carle, S. Zander and T. Zseby, "Policy-based Accounting", RFC 3334, October 2002

[7] R. Kühne et al.: "Architecture for a Service-Oriented and Convergent Charging in 3G Mobile Networks and Beyond", 6th IEE International Conference on 3G & Beyond, 2005

[8] 3GPP TS 23.125: "Overall high level functionality and architecture impacts of flow based charging", Release 6