Adaptive Resource Control for QoS
Using an IP-based Layered Architecture

Project Review No. 3
Dresden, Germany
November 21 - 23, 2001

http://www.ist-aquila.org/
Outline

- Project Overview
  - Bert F. Koch (Siemens)

- Measurement Architecture for Development and Operation of DiffServ Networks
  - Gerald Eichler (T-Systems Nova)
  - Ulrich Hofmann (Salzburg Research)

- Control Loops
  - Thomas Engel (Siemens)

- BGRP Quiet Grafting: An Approach for a Scalable Inter-Domain Resource Control
  - Martin Winter (Siemens)
Architecture

Resource Control Layer

Resource Control

Consideration of Network Load

Monitoring Probing Results

Admission Control

End-user Application Toolkit

QoS Request

QoS Request

Setting

Setting

Access Network

Access Network
Measurement Architecture for Development and Operation of DiffServ Networks
Outline

- Measurement architecture
- Measurements within the 1st trial period
- Operator friendly GUI
- Improved load generators
- Passive measurement for MBAC validation
- State of the art, research and development, exploitation
AQUILA - Why Measurements?

**IntServ Model**
- RSVP
- end-to-end QoS
- Soft state
- Absolute

**DiffServ Model**
- Per hop behaviour
- CoS per link
- Stateless
- Relative

Focus
- QoS
- System view
- Guarantees

QoS Proven

?
AQUILA - Measurement Focus

1st trial
- Proof of RCL architecture

Measurements

2nd trial
- Measurement Based Admission Control
- Resource Pool (Re)Allocation

SLS validation
Measured Parameters

- **Performance parameters**
  - One way delay (OWD) → accuracy 200 µs
  - Instantaneous Packet Delay Variation (IPDV) → accuracy 200 µs
  - Packet loss
  - Throughput

- **Router statistics**
  - Traffic rates
  - Queue lengths
  - Packet drop counters
Measurement Architecture for the 1st Trial

MAa – Measurement Agent (application-like)
MAp – Measurement Agent (probing)
MIC – Management Information Collector
DB – Database

H – Host
ER – Edge Router
CR – Core Router

User Flow
Measurement Flow
Components of the Distributed Measurement Architecture (DMA)

- **MAa**: Measurement agents with traffic generators that produce application-like flows (reproducible experiments as opposed to real applications)
- **MAP**: Measurement agents that perform active network probing (constant monitoring of the whole network)
- **MIC**: Management Information Collectors that supply router statistics (traffic rates, queue lengths, packet drop counters etc.)
- **DB**: Database for configuration data and measurement results
Outline

- Measurement architecture
- **Measurements within the 1st trial period**
- Operator friendly GUI
- Measurements for MBAC
- Improved load generators
- Passive measurement for MBAC validation
- State of the art, research and development, exploitation
DMA - Reference Trial Site (Warsaw)

- **4 Measurement Agents (MA 1 - 4)**
  - GPS synchronised
  - MAa and MAP co-located

- **MIC and server components co-located**
DMA - Example Measurement

One Way Delay (OWD) as a function of packet number

![Diagram showing network topology and One Way Delay measurements](image-url)
DMA - Feedback and Improvements

- Lessons learnt from 1\textsuperscript{st} trial
- New requirements
  - Navigation ✓
  - DMA configuration ✓
  - Online-monitoring of selected parameters ✓
  - Usability ✓
  - Stand-alone operation

⇒ Implementation in the 1\textsuperscript{st} trial extension and 2\textsuperscript{nd} trial
Outline

- Measurements within the 1st trial period
- **Operator friendly GUI**
- Measurements for MBAC Measurement architecture
- Improved load generators
- Passive measurement for MBAC validation
- State of the art, research and development, exploitation
DMA - Navigation Improvement

- Explorer-like navigation menu
- Re-ordering of the menu items
- Reduced number of menu items
DMA - Configuration Quickstart

- Configuration assistant
- Leads through the configuration steps
- Easy to configure fully meshed measurement scenarios
DMA - Online Monitoring

- Per traffic class
- Graphical display of measurement results during running tests
  - One Way Delay
  - Delay variation
  - Packet loss
  - Throughput
DMA - Measurement Agent Controller

- Easy to use Measurement Agent Controller (MACON)
  - Controls up to 10 measurement agents
  - 5 traffic classes
  - Automatic start of fully meshed measurements
  - Online-monitoring of the system status
  - Online-monitoring of one-way delay, delay variation and packet loss
  - Data export to csv-files

MA : measurement agent
MA 1 → MA 2
MA 1 → MA n
MA 2 → MA n
MA 1 → MA 3
MA 2 → MA 3
MA 3 → MA n

MACON

configuration/results
measurement flow
DMA - Easy to Use GUI

**Functionality**

- Easy configuration
- Database independence
- Fast agent selection
- Fast traffic selection
- Integrated online-monitoring
Outline

- Measurement architecture
- Measurements within the 1st trial period
- Operator friendly GUI
- Measurements for MBAC
- Improved load generators
- Passive measurement for MBAC validation
- State of the art, research and development, exploitation
DMA - Control Loop Support (1)

- Resource control in the 1st trial (open loop)
DMA - Control Loop Support (2)

- **Resource control in the 2\textsuperscript{nd} trial (closed loop)**
  - Measurement Based Admission Control (MBAC)
  - Provisioning Control Loops (PCL)
DMA - OWD Measurements for MBAC

- Is the measured One Way Delay suitable for Measurement Based Admission Control (MBAC)?
MBAC Implementation: Function Split

- **MbacMonitor**
  - Measurement agent attached to each ACA
  - Retrieves measurement data $X_{(i)}$ from edge routers
    - manages a list of router interfaces which are to be monitored
    - polls edge router at the end of each measurement interval
    - either monitoring of transmitted bytes on input ports
    - or monitoring of transmitted and dropped bytes on output ports

- **MbacProcessor**
  - Estimates mean load $M_{est(i)}$ from measurement data
  - Keeps state information (mean load, PRs, ...)
    - for each ED, traffic class and direction (ingress, egress)
  - Implements MBAC algorithms
  - AC decision
DMA - Router Statistics for MBAC

- Admission control loop using statistics data from routers
  - Monitor real traffic load on edge link(s) to perform MBAC
  - Retrieve traffic rates for both outgoing and incoming traffic from the router
  - Store traffic rates for MBAC algorithm processing

![Diagram of resource control layer with MbacMonitor and MbacProcessor nodes connected to ED1, ED2, ED3, and CR nodes.]
DMA - Router Statistics

- **Management Information Collector enhancements for 2nd trial**
  - Distributed design
    - agent controller and DMA database interface
    - Edge Router monitoring integrated with corresponding Admission Control Agents
      » less interference to network traffic due to monitoring
      » agents store results to ACA for MBAC and optionally to the DMA database
    - core router monitoring agents stay co-located with the measurement server
  - Easier configuration of router measurements
Outline

- Measurement architecture
- Measurements within the 1st trial period
- Operator friendly GUI
- Measurements for MBAC
- Improved load generators
- Passive measurement for MBAC validation
- State of the art, research and development, exploitation
Virtual WEB User (VWU)

- **3 components**
  - WEB Proxy → traces {object_size, think_time, ...}
  - Statistical models for objects, think_times
  - Load generator: models, traces

- **AQUILA: trial integration of VWU (3/02)**
Virtual WEB User Testing

- **WEB tests with more realistic load generators**
  - Derived from different application and network scenarios
  - Self-adaptive to network conditions (traces of 64 kbps link users are different from 2 Mbps link users)
  - New load generator objects

- **QoS-test for different user characteristics**

---

![Diagram showing user characteristics](image)

user1: long active/inactive intervals

user2: short active/inactive intervals
Perceptual VoIP QoS Measurement (1)

- **From „end-to-end“ to „mouth-to-ear“**
- **Perceptual VoIP QoS measurements**
  - e2e IP-QoS $\Rightarrow$ e2e application QoS (PCBR, TCL1)
    - IP {one_way_delay, jitter, loss} $\Rightarrow$ mouth_to_ear QoS
  - Generation of voice samples (e.g. *.wav-files)
  - Transmission, Perceptual QoS; IPQoS $\Leftrightarrow$ Perceptual QoS

```
 Soundcard, Perceptual QoS

\[\text{voice} \rightarrow \text{Codec} \rightarrow \text{RTP/RTCP} \rightarrow \text{UDP, IP}\]
```

\[\text{voice} \leftarrow \text{Codec} \leftarrow \text{RTP/RTCP} \leftarrow \text{UDP, IP}\]
Perceptual VoIP QoS Measurement (2)

- **Implementations:**
  - PAMS Perceptual Analysis Measurement System (BT 1998)
  - PSQM, PSQM+ Perceptive Speech Quality Measurement (BT, KPN)
  - TOSQA Telecommunications Objective Speech Quality Assessment (DT)
  - Products HP: Agilent Telegra VQT100, $39,000, monitoring of alternative routing for IP telephony

- **Standards:**
  - ITU-T Rec. P.57: Artificial Ear
  - ITU-T Rec. P.862: Perceptual Evaluation of Speech Quality, an Objective Method for end-to-end Speech Quality Assessment of Narrow Band Telephone Networks and Speech Codecs
Perceptual VoIP QoS Measurement (3)

- **QoS = MOS (Mean Opinion Score)**
  - 5 = Excellent; 4 = Good; 3 = Fair; 2 = Poor; 1 = Bad; 0 = ----
  - Test scenario:
    - VoIP-TCL1-Admission (e.g. BW = 10 kbps)
    - transmission
    - results:
      » IP-QoS: Delay, Loss
      » MOS

- **Research project: correlation IP-QoS ⇔ MOS**
Measurement Process

1. Test Editing via WWW-Browser
   - new: choose voice sample

2. Storing of configuration data to the database
3. Retrieving configuration data from the database
4. Distribution to Measurement Agents (MA)
   - mgmt process

5. Executing test result process

6. Reporting results

7. Storing results

8. Browsing results with WWW-Browser
   - new: MOS diagram

new: sample generation / compression / decompression
QoS MOS analysis
Outline

- Measurement architecture
- Measurements within the 1st trial period
- Operator friendly GUI
- Measurements for MBAC
- Improved load generators
- **Passive measurement for MBAC validation**
- State of the art, research and development, exploitation
Approaches for MBAC Measurements

- AQUILA QoS target: loss rate < $10^{-4..-6}$ | max #flows
- Why don’t we measure loss? Rare events!
  - Difficult to measure for operational purposes
- What to measure?

M. Siler, J. Walrand, IWQoS98
1\textsuperscript{st} step: measurement based estimation of buffer occupancy

AQUILA
1\textsuperscript{st} step: measurement based estimation of link load

model: buffer occupancy $\Leftrightarrow$ loss

model: link load $\Leftrightarrow$ loss

\textbf{2\textsuperscript{nd} step: derivation of loss probabilities}
Measurement Interval (1)

Open questions: what to measure
measurement intervals [µs ... s ... min]
Measurement Interval (2)

**Validation necessary**

- Measurement interval small enough?
- No: high load values smoothed $\Leftrightarrow$ loss intervals not detected
- Yes:
  - bucket parameters = worst case variation
  - long term variations can be measured
    » validation of the AQUILA MBAC by passive and active measurements
Validation Architecture

- **Accurate variance estimation: Passive measurements**

  - Validation #flows_router=#flows_DAG ?
  - AQUILA RCL Layer variance, loss [n*100ms] → #flows
  - PC
  - DAG
  - Splitter: ATM, SDH
  - Router

  100 % capturing: up to 2 Gbps
Loss Rate Estimation

Sequential testing [Wald]
- Hypothesis 1: $p(\text{loss}) = p_1$
- Hypothesis 2: $p(\text{loss}) = p_2$
- $\alpha$: type 1 error

**Result for $10^{-3}$**
- Mean = 1000 packets $\Rightarrow$ take 2000 packets
- $\alpha$-accurate loss estimation for passive and active measurements
- Estimation of test run length (new feature DMA edit GUI 2nd trial)
DMA Status and Plans

- **Operational measurements (Control Loop)**
  - Router statistics (load, future R&D: loss)
  - Probing

- **Validation measurements**
  - Application-like flow generators
  - Passive measurements with DAG card (load, loss)
  - Router monitoring (load, loss)

- **Continuously enhanced GUI, DataBase**
Outline

- Measurement architecture
- Measurements within the 1st trial period
- Operator friendly GUI
- Measurements for MBAC
- Improved load generators
- Passive measurement for MBAC validation
- State of the art, research and development, exploitation
State of the Art - IETF (1)

- Traffic flow measurement architecture
  - RFC2722 [N. Brownlee, C. Mills, G. Ruth, October 1999]
  - Today's measurement architectures: how, what?
- Next steps for the IP QoS architecture
  - RFC2990 [G. Huston, Nov. 2000]
  - QoS discovery
    - no mechanisms exist to query the network for the potential to support a specific service profile
  - QoS Routing and Resource Management
    - spreading the load across a broader collection of network links
  - Calculate per-path dynamic load metrics
  - Metric: path's potential to carry additional traffic
State of the Art - IETF (2)

- **A Migration Path to provide End-to-End QoS over Stateless Networks by Means of a Probing-driven Admission Control**
  - `draft-bianchi-blefari-end-to-end-qos-01.txt`, 7/2001
  - Implicit signalling paradigm GRIP (Gauge&Gate Reservation with Independent Probing)
  - Probes and Information packets different labels (DS codepoint) service priority to Information packets

- **Transport Performance Metrics MIB**
  - `draft-ietf-rmonmib-tpm-mib-03.txt`, R. Dietz, R. Cole 7/2001
  - General framework for the collection and reporting of performance related metrics on traffic flows in a network
State of the Art - Conferences (1)

**PAM (Passive and Active Measurement): 2001 Amsterdam**

- Telcordia RONDO: Real-Time Measurement → MPLS re-routing
- SPRINT: Passive Measurement DAG → 5µs accuracy; 16 LINUX-Cluster
- RIPE: Active Measurements, Infrastructure

30 s pathological behaviour: difficult to detect without monitoring
State of the Art - Conferences (2)

**SIGCOM IMW2001**

- J. Micheel, S. Donelly, I. Graham: Precision Timestamping of Network Packets
- G. Iannaccone, C. Diot, I. Graham, N. McKeown: Monitoring very high speed links
State of the Art - Conferences (3)

- Shenker et al.
  - No difference between the different MBAC algorithms
  - Network operators will need to monitor actual performance in order to learn appropriate parameter setting for prediction, smoothing, ...

![Graph](image)
Exploitation

- Bilateral projects with AQUILA partners Siemens, TA, DT
- Courses at FH Salzburg (Internet-Technologies)
- Interdisciplinary research: Telemedicine
- New EU IST proposal
CeBIT 2002

- 13.-20.3. 2002 Hall 11
  FH Salzburg & SalzburgResearch

- AQUILA demo (1st concept)
  - 3 „real“ applications:
    PCBR, PVBR, STD
  - Increase background load
  - Show QoS stability for
    PCBR, PVBR

- AQUILA documentation
Control Loops
Outline

- Overview
- Measurement Based Admission Control (MBAC)
- Provisioning Control Loops (PCL)
Resource Management in 1st Trial

- Traffic forecasts
- Measurements
- Traffic models
- Traffic descriptor

Provisioning

Resource Pool

Admission Control

Individual Flow

View:

- Global network
- Traffic collection / distribution area near network border
- Available bandwidth at edge router
- Perceived QoS
Control Loops (1)

**Idea**

- Use feedback from measurements
  - traffic load
  - QoS
- Adapt resource management to real situation
- Instead of blind following assumptions
  - traffic forecasts
  - traffic models
Control Loops (2)

- traffic forecasts
- traffic models
- traffic descriptor
- Provisioning
- Resource Pool
- Admission Control
- individual flow
- MBAC
- PCL

measurement
measurement
Outline

- Overview
- Measurement Based Admission Control (MBAC)
- Provisioning Control Loops (PCL)
MBAC Approaches

- **Aggregate stream measurement**
  - Estimation of mean rate
    - simple implementation
  - Estimation of mean rate and variance
    - variability of traffic is difficult to capture
    - some self-similar traffic models show unlimited variance
    - requires special measurement functions
    - requires very small measurement intervals,
      if variance has to be determined from mean rate measurements
    - better performance in terms of achieving QoS targets

- **Per flow measurement**
  - Complex implementation
MBAC Approaches

**Aggregate stream measurement**
- Estimation of mean rate
  - simple implementation
- Estimation of mean rate and variance
  - variability of traffic is difficult to capture
  - some self-similar traffic models show unlimited variance
  - requires special measurement functions
  - requires very small measurement intervals, if variance has to be determined from mean rate measurements
  - better performance in terms of achieving QoS targets

**Per flow measurement**
- Complex implementation
Estimation of Mean Rates

**Sampling**
- Sampling intervals with fix length $T$
- $X(i)$ - measured mean rate in sampling interval $i$

**Estimation**
- Moving average (window based mean rate estimation – moving window)
- Fix window size $K$ (number of sampling intervals)
- Mean of measurement values of $K$ sampling intervals

$$M_{est}(i) = \frac{1}{K} \sum_{j=0}^{K-1} X(i - j)$$

- After each sampling interval a new $M_{est}$ is calculated
Measurement Scheme

\[ M_{est}(i) = \frac{1}{K} \sum_{j=0}^{K-1} X(i - j) \]
Traffic Classes

- **Five Traffic Classes have been specified**

<table>
<thead>
<tr>
<th>Network service</th>
<th>Premium CBR</th>
<th>Premium VBR</th>
<th>Premium MultiMedia</th>
<th>Premium Mission Critical</th>
<th>Standard</th>
</tr>
</thead>
<tbody>
<tr>
<td>Traffic class</td>
<td>TCL 1</td>
<td>TCL 2</td>
<td>TCL 3</td>
<td>TCL 4</td>
<td>TCL STD</td>
</tr>
</tbody>
</table>

- **... as well as the related Traffic Control Mechanisms in the Routers**
MBAC for TCL1
Basic Scheme

\[ PR_{new} + M_{est} \leq \rho C_1 \]

- **Aggregate sum**
- \( PR_{new} \): peak rate of new flow
- \( M_{est} \): estimation of aggregate mean rate
- \( \rho \): utilisation target (tuneable parameter)
- \( C_1 \): resources available for TCL1 (e.g. AC limit)
MBAC for TCL1
Peak Rate Reservation Refinement

\[ PR + M_{est} \leq \rho C_1 \]

\[ PR = PR_{new} + \sum_{i=1}^{A} w_i \cdot PR_{aggr}^i \]

\[ PR_{aggr}^i = \sum_{j=1}^{n_i} PR_{ij} \]

\[ w_i = e^{-i/\tau} \]

- Takes peak rates of previously accepted flows into account
  - recently admitted flows are reflected in measurements after some delay only
  - avoids accepting more flows than fit into the available bandwidth
- \( PR_{aggr}^i \) sum of PR of flows accepted in measurement interval \( i \)
- \( A \) aging window
- \( PR_{ij} \) PR of \( j^{th} \) reservation in measurement interval \( i \)
MBAC for TCL2

\[ PR_{\text{new}} + M_{\text{est}} + \sqrt{\frac{\gamma}{2} \sum_{i=1}^{N_2} PR_i^2} \leq C_2 \]

\[ \gamma = -\ln(P_{\text{loss}}) \]

- **Hoeffding bound**
- \( PR_{\text{new}} \) - peak rate of new flow
- \( M_{\text{est}} \) - estimation of aggregate mean rate
- \( C_2 \) - resources available for TCL2 (e.g. AC limit)
- \( N_2 \) - number of reservations in TCL2
- \( PR_i \) - peak rate of \( i^{th} \) still active reservation
- \( P_{\text{loss}} \) - target packet loss ratio
MBAC for TCL3

\[
( SR_{\text{new}} + \sum_{k} SR_k \leq \rho_3 C_3 ) \quad \text{or} \quad ( SR + M_{\text{est}} \leq \rho_3 C_3 )
\]

\[
SR = SR_{\text{new}} + \sum_{i=1}^{A} w_i \ SR_{i}^{\text{aggr}} \quad SR_{i}^{\text{aggr}} = \sum_{j=1}^{n_i} SR_{ij} \quad w_i = e^{-i/\tau}
\]

- EITHER: sum of SRs has to be below the available capacity
- OR: similar MBAC method as used for TCL1 base on
  - sustainable rate
  - estimation of mean rate of „in-profile“ packets

The sum of declared SR parameters cannot over-allocate the available capacity more than n-times (protection against measurement error)

- \( SR_{i}^{\text{aggr}} \) sum of SR of flows accepted in interval i
- A aging window
Potential Benefits of MBAC (1)

- Token bucket characterisation requires significant over-allocation of token rate in case of real traffic sources
- Required SR is significantly larger than mean rate
  - 3 times for $P_{loss} = 10^{-4}$

![Diagram showing packet loss vs. SR/m for different BSS values](image)
Potential Benefits of MBAC (2)

MBAC is more efficient than DBAC (Declaration Based Admission Control) when SR/m is > 2

![Diagram showing link utilisation for MBAC versus DBAC with different token bucket parameters](image-url)

- MPEG trace
- Different token bucket parameter - token rate
Outline

- Control Loops
- Measurement Based Admission Control (MBAC)
- Provisioning Control Loops (PCL)
Does AQUILA need a PCL?

■ What bandwidth is needed between CR5 and CR1?
  • Routing minimises hop count
  • All AC limits are 1 Mbps
  • Equal traffic distribution: 0.6 Mbps
  • Worst case: 2 Mbps

■ How is blocking probability?
PCL Aims

- **Dynamic control of**
  - Scheduling (WFQ weights)
  - AC limits
  - RP limits

- **Goals**
  - Optimised resource utilisation
  - Assure QoS targets

- **Constraints**
  - There is sufficient BW
  - Adapt BW partition for traffic classes only
PCL Architecture

- data collection
- re-provisioning
- re-configuration

- measurements
- demand estimations
- notifications

Diagram showing the interconnections between RCA and ACA nodes, with arrows indicating data flow.
PCL Sequence of Events

- **Measurement of blocking frequency**
  - At each ACA for each (TCL, ER)

- **Estimation of BW demand**
  - DBAC: at each ACA for each (TCL, ingress ER, egress ER)
  - MBAC: at each ER for each (TCL, interface)

- **Problem detection**
  - Notification triggers re-provisioning

- **BW partition**
  - For each (TCL, link)

- **Computation provisioning parameter**
  - WFQ weights for ER, CR
  - AC limits for each (TCL, ER)
  - RP limits for each RP
Demand Measurement and Estimation

Demand Estimation: $\tilde{a}_{ij}(t,s)$

- Reserved BW $r(t,s,i,j)$
- Rejection Rate $b(t,i)$
- Reserved BW $r(t,s,i)$

Traffic Descriptor

DBAC

MBAC

Measurement

ER
DBAC Based Re-Provisioning

- **Event triggered**

  ![Graph showing event triggered re-provisioning](image)

  \[ \begin{pmatrix} \tilde{a}_{11}(t,s) & \tilde{a}_{12}(t,s) & \cdots & \tilde{a}_{1N}(t,s) \\ \tilde{a}_{21}(t,s) & \tilde{a}_{22}(t,s) & \cdots & \tilde{a}_{2N}(t,s) \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{a}_{N1}(t,s) & \tilde{a}_{N2}(t,s) & \cdots & \tilde{a}_{NN}(t,s) \end{pmatrix} \leq \begin{pmatrix} a_{11}(s) & a_{12}(s) & \cdots & a_{1N}(s) \\ a_{21}(s) & a_{22}(s) & \cdots & a_{2N}(s) \\ \vdots & \vdots & \ddots & \vdots \\ a_{N1}(s) & a_{N2}(s) & \cdots & a_{NN}(s) \end{pmatrix} \]

- **Local observation of demand matrix at network border**
MBAC Based Re-Provisioning

- **Event triggered**

  ![Event triggered diagram]

- **Local observation of demand vector at network border**

  \[
  \begin{pmatrix}
  \tilde{A}_1(t,s) \\
  \tilde{A}_2(t,s) \\
  \vdots \\
  \tilde{A}_N(t,s)
  \end{pmatrix}
  \leq
  \begin{pmatrix}
  A_1(s) \\
  A_2(s) \\
  \vdots \\
  A_N(s)
  \end{pmatrix}
  \]}
Summary

- **Assumptions are replaced by measurements**
  - Improved resource utilisation / QoS

- **Local measurements at the network edge**
  - Window based mean rate estimations at edge router interfaces
  - Demand and blocking in ACAs

- **Local processing of measurement data at the network edge**
  - ACA

- **Different MBAC algorithms for TCL1, TCL2 and TCL3**
  - Adapted to specific requirements of each traffic class

- **Next steps**
  - Implementation and test of control loops in second phase
  - Performance studies
  - Validation by passive measurements
BGRP Quiet Grafting

An Approach for a Scalable Inter-Domain Resource Control
Outline

- Short Review of BGRP
- BGRP and scalability: problems and solutions
- Simulation results
Short Review of BGRP

- **Inter-domain resource reservation protocol**
  - Additional layer on top of intra-domain resource control

- **BGRP agents associated with border routers**
  - May run co-located or on other associated device

- **Depends on BGP running on the BRs**
  - Uses the BGP sink trees for reservation aggregation

- **Simple soft-state messages**
  - Downstream: PROBE message identifies sink tree and checks policy
  - Upstream: GRAFT message makes reservation
Short Review of BGRP

Domain may use other QoS mechanisms than AQUILA

Depends on BGP running on the BR
Outline

- Short Review of BGRP
- BGRP and scalability: problems and solutions
- Simulation results
BGRP and Scalability

- **BGRP aggregates reservations along BGP sink trees**
- **BGRP limits the number of active reservations at each node**
  - Maximum: number of Autonomous Systems (AS) in the Internet
  - BGRP limits memory usage in BGRP agents
- **BGRP does not reduce the number of signalling messages**
  - PROBE and GRAFT messages travel end-to-end
  - BGRP does not limit the necessary processing power in BGRP agents
Quiet Grafting

**How can we reduce the number of signalling messages?**
- Shorten the path of the messages
- Answer PROBE messages as early as possible before the destination AS

**What is necessary for quiet grafting? Three problems!**
- Identify the sink tree before the destination AS
- Use pre-reserved resources on the sink tree
- Establish mechanisms for reservation of resources within the destination domain
Problem 1: Identification of Sink Tree

- **Sink tree identification**
  - Sink trees are identified by their root: the destination AS number and an identification of the border router in the destination AS (entry point)
  - Sink trees collect all traffic identified by the Network Layer Reachability Information (NLRI) announced by that root

- **Sink trees are not BGP routes**
  - BGP may aggregate several routes
  - However, multiple BGRP sink trees must not be aggregated
  - So BGP routing information is not sufficient to identify a sink tree
Solution 1: BGRP Sink Tree Identification

- **Install sink tree identification information**
  - Return sink tree identification information with GRAFT messages
    - AS, BR id
    - NLRI
  - Store this information at every BGRP agent, which handles the GRAFT message

- **Use information for early sink tree identification**
  - Match further requests with the stored NLRI to identify the sink tree as early as possible
  - Sink tree may be identified at the point, when a PROBE message hits an already existing reservation for that sink tree
  - Use REFRESH messages to validate the NLRI information
BGRP Sink Tree Identification

- Installing the NLRI information
  - First PROBE sent from AS2 to AS5
  - GRAFT installs AS5 NLRI in intermediate BGRP agents

- Use NLRI for quiet grafting
  - PROBE from AS1 hits the sink tree at the egress BR of AS4
  - AS4 may already return a GRAFT message without forwarding the message to AS5
Problem 2: Pre-Reserved Resources

- **Need for resource cushions**
  - A GRAFT message can only be generated, when pre-reserved resources are available within the network

- **Trade-off between network utilisation and signalling load reduction**
  - It is a crucial point for network utilisation to find a proper algorithm
Solution 2: Delayed Resource Release

- **Regular checks**
  - Check resource utilisation at regular intervals

- **Return policy**
  - Return fixed size block of resources, if it was unused during the complete Retain Period (RP)
  - Return Block Size (RBS) and Retain Period are configurable and determines performance of algorithm

![Diagram showing resource utilisation and release over time]
Problem 3: Reservation in the Last Domain

- **Destination domain is not aware of new reservation**
  - With quiet grafting, there is no signalling message travelling to the destination domain, which indicates the new reservation
  - Resource availability is guaranteed throughout the way to the destination domain, but not within that domain
Solution 3: Direct Signalling

- **Establish communication**
  - Along with the GRAFT message, return an object reference to the intra-domain resource control
  - Store this object reference along with the sink tree identification and NLRI at each intermediate BGRP agent

- **Direct interaction**
  - Use direct communication between the source domain and the destination domain to establish a reservation in the destination domain
  - No signalling load for intermediate domains
Signalling to the Last Domain

Domain may use other QoS mechanisms than AQUILA.
Will This Work?

- **Sink tree identification**
  - How early can we identify a sink tree?

- **Resource cushion**
  - How often can we re-use resources from a resource cushion?

- **Answers depend on many parameters**
  - Network topology
  - Traffic pattern
  - Configuration of algorithms
Outline

- Short Review of BGRP
- BGRP and scalability: problems and solutions
- Simulation results
Proof of Concept by Simulations

- **Earliest point for quiet grafting**
  - How far has a PROBE message to travel, until we can identify the corresponding sink tree?

- **Effectiveness of resource cushions**
  - Are resource cushions available where needed?
  - What is the overhead of unnecessarily reserved resources?
Earliest Point for Quiet Grafting

- Some definitions
  - Sink tree: exists independently of actual reservations
  - Populated node: node with an actual resource reservation for that sink tree

- Basic considerations
  - In a sparsely populated sink tree, the average distance from a node to the first populated node is rather high
  - In a densely populated sink tree, the average distance from a node to the first populated node is rather low
  - As the population of the tree increases, the average distance to the first possible point for quiet grafting decreases
First Results

Sink Tree of depth 10

![Sink Tree of depth 10](image)
Effectiveness of Resource Cushions

- **Build resource cushions through delayed resource release**
  - Released resources are not immediately forwarded towards the sink tree root, but used to build a resource cushion

- **Release unused resources**
  - Resource cushions are released step-wise (Retain Period) and block-wise (Return Block Size)
  - When more than RBS resources are unused during a RP, then this block is released
First Simulation Results (1)

**Parameters**
- Traffic generator with exponential distributed inter-arrival time and holding time
- Request size = 1 u
- Mean offered load = 20 u
- Mean holding time = 3 min
- RP = 5 min
- RBS = 1 u

**Results**
- 30% average resource cushion
- 97% of requests served by cushion
First Simulation Results (2)

- **Variation of Retain Period**
  - Mean offered load = 20 u
  - Mean offered load = 100 u

- **Results**
  - R = requests forwarded
  - C = mean cushion size

- **Trade-off**
  - Higher RP yields lower R but higher C

- **Load dependent**
  - Higher load gives better results

![Graph showing variation of retain period]
Conclusions on First Results

- **Proof of concept**
  - Delayed resource release can dramatically reduce the signalling load
  - Reasonable mean size of resource cushions (10% - 30%)

- **Especially effective in lively branches of the sink tree**
  - Best results when number of incoming requests is high

- **Optimal parameter sets will vary at different places in the sink tree**
  - Short RP and small RBS near the leafs
  - Longer RP and larger RBS near the sink
AQUILA (IST-1999-10077)

Adaptive Resource Control for QoS
Using an IP-based Layered Architecture

Project Review No. 3
Dresden, Germany
November 21 - 23, 2001

Thank you for your attention!
http://www.ist-aquila.org/