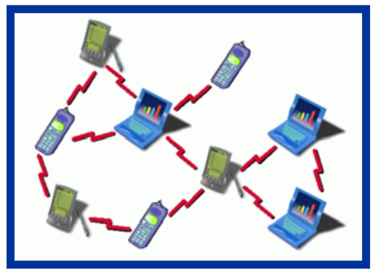
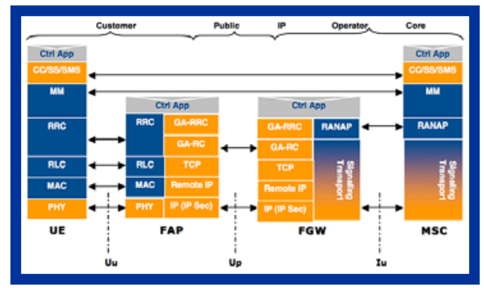
Industrial Application of Coloured Petri Nets for Protocol Verification





Lars M. Kristensen

Department of Computer Engineering

Bergen University College, NORWAY

Email: lmkr@hib.no / Web: www.hib.no/ansatte/lmkr

Practical Applications

- **CPNs and state space methods have been widely used for protocol verification purposes:**
  - Danfoss Flowmeter Systems.
  - Bang & Olufsen Beolink System.
  - Ericsson Edge Router Discovery Protocol.
  - Several Internet protocols
    (e.g., WAP, IOTP, TCP, DCCP, SIP, DYMO).
  - ...
- **For a comprehensive list of examples, see:**

  http://www.cs.au.dk/CPnets/intro/industrial.shtml

Overview

- **Two examples of industrial application of CPN technology for protocol verification.**
- **Specification and Validation of an Edge Router Discovery Protocol for Mobile Ad-hoc Networks:**

L.M. Kristensen and K. Jensen. *Specification and Validation of an Edge Router Discovery Protocol for Mobile Ad Hoc Networks.* In Integration of Software Specification Techniques for Applications in Engineering, pages 248-269. Volume 3147 of Lecture Notes in Computer Science. Springer, 2004

BAE SYSTEMS

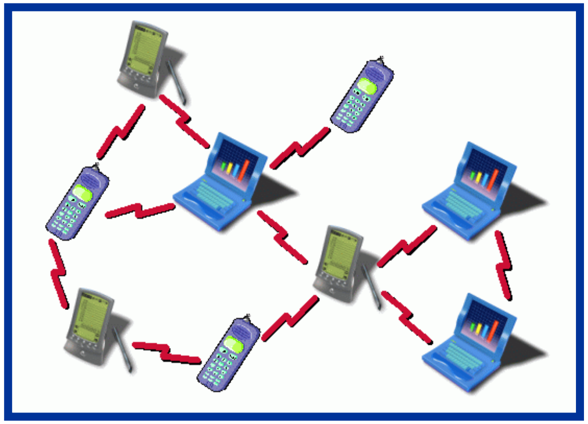
ERICSSON

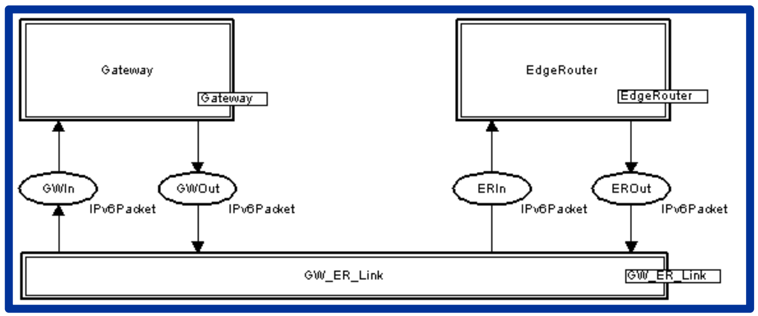
- **Formal Specification and Validation of Secure Connection Establishment in a Generic Access Network Scenario:**

P. Fleischer and L.M. Kristensen. Modelling and Validation of Secure Connection Establishment in a Generic Access Network Scenario. In Vol. 94, No. 3-4 of Fundamenta Informaticae, pp. 361-386, IOS Press, 2009.

TietoEnator

Specification and Validation of an Edge Router Discovery Protocol for Mobile Ad Hoc Networks



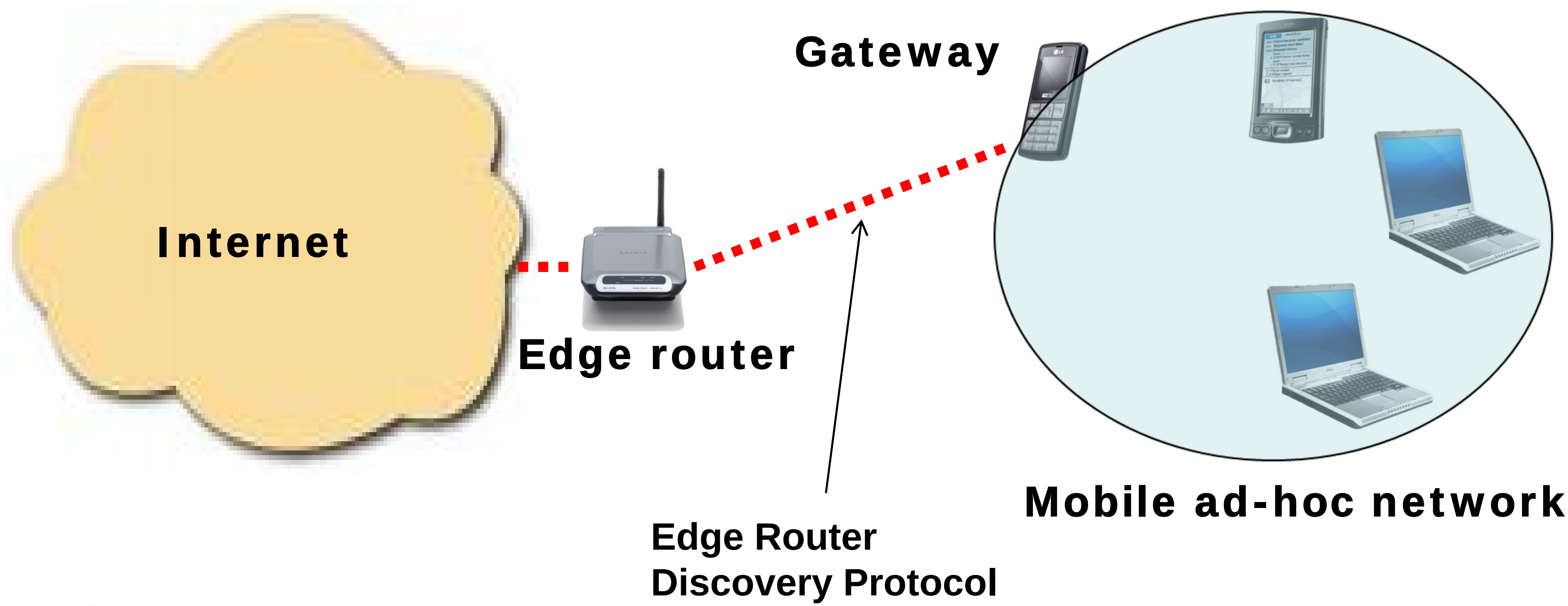


Project Aims and Setup

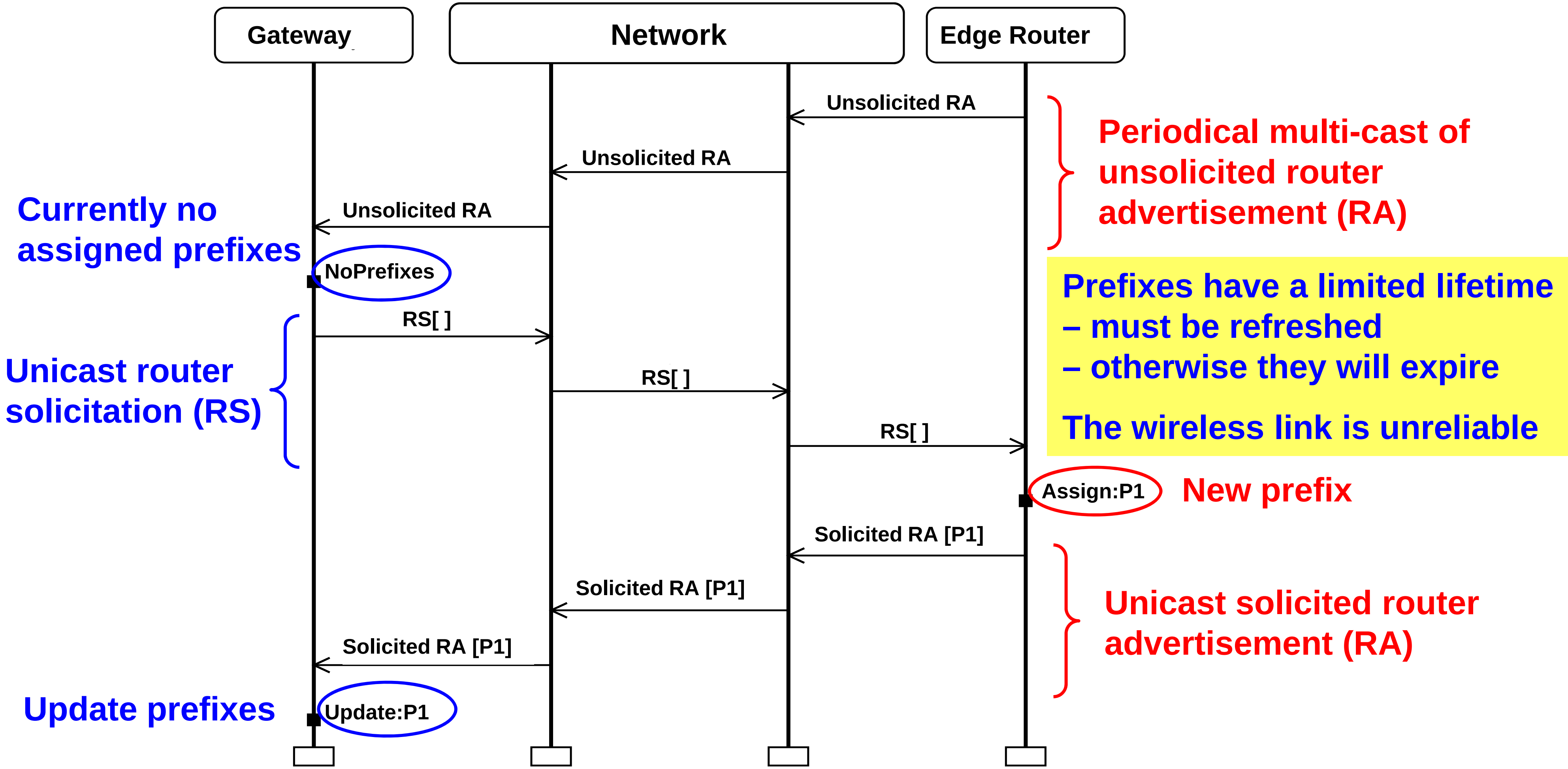
- **Project context:**
  - Development of the Edge Router Discovery Protocol (ERDP) for MANETs based on the IPv6 NDP Protocol.
  - Apply of Coloured Petri Nets (CPNs) and CPN Tools in the development of protocol software.
  - The software engineers were given a 6-hours course on CPN modelling.
- **Application of CPN technology:**
  - A CPN model was constructed constituting a formal specification of the ERDP protocol.
  - State space exploration was applied to conduct a formal verification of key properties of ERDP.
  - Modelling and verification helped in identifying several omissions and errors in the design.

Edge Router Discovery Protocol

- **Protocol for IPv6 prefix configuration executed between edge routers and gateways:**



Configuration of a gateway

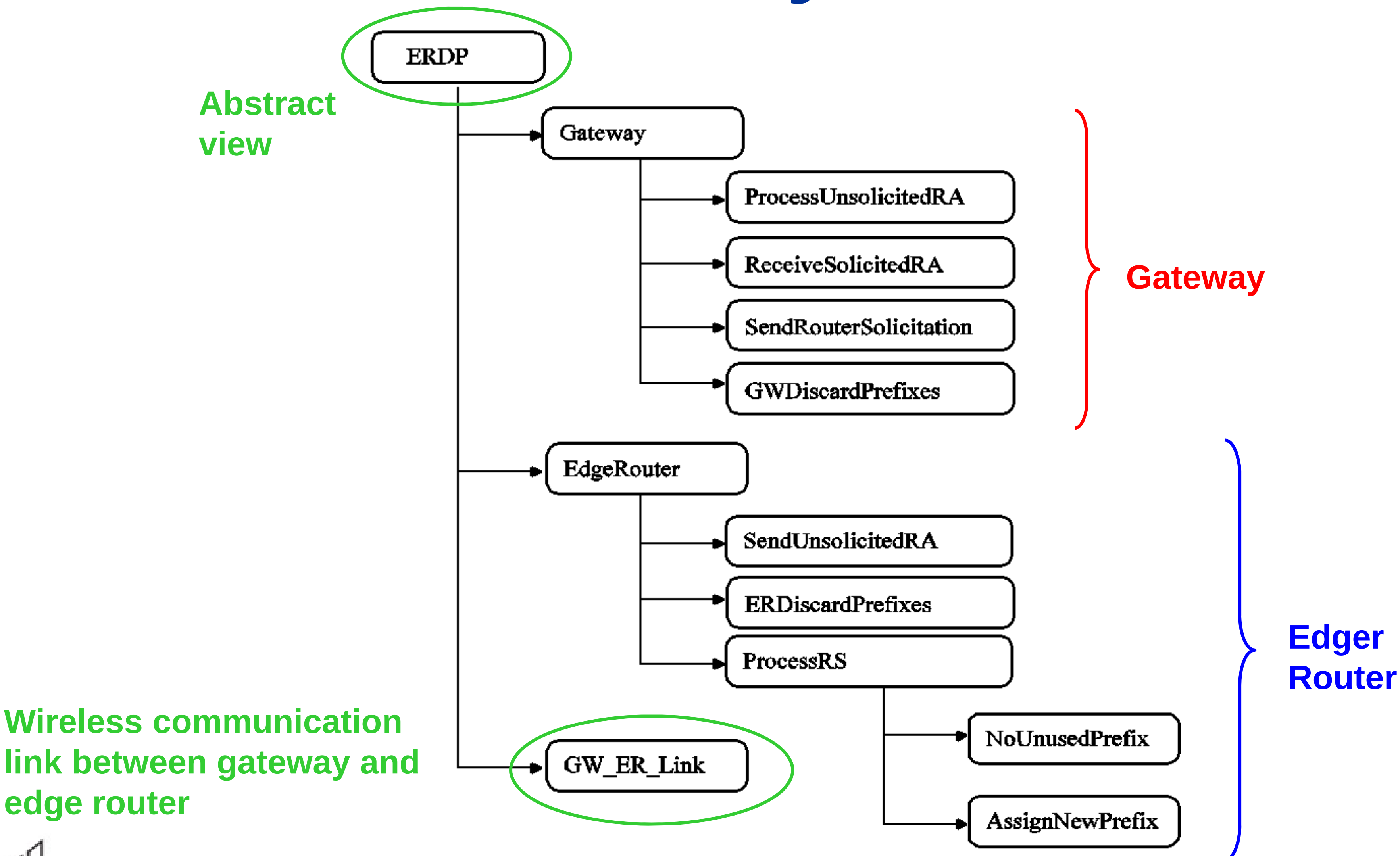


BERGEN UNIVERSITY COLLEGE

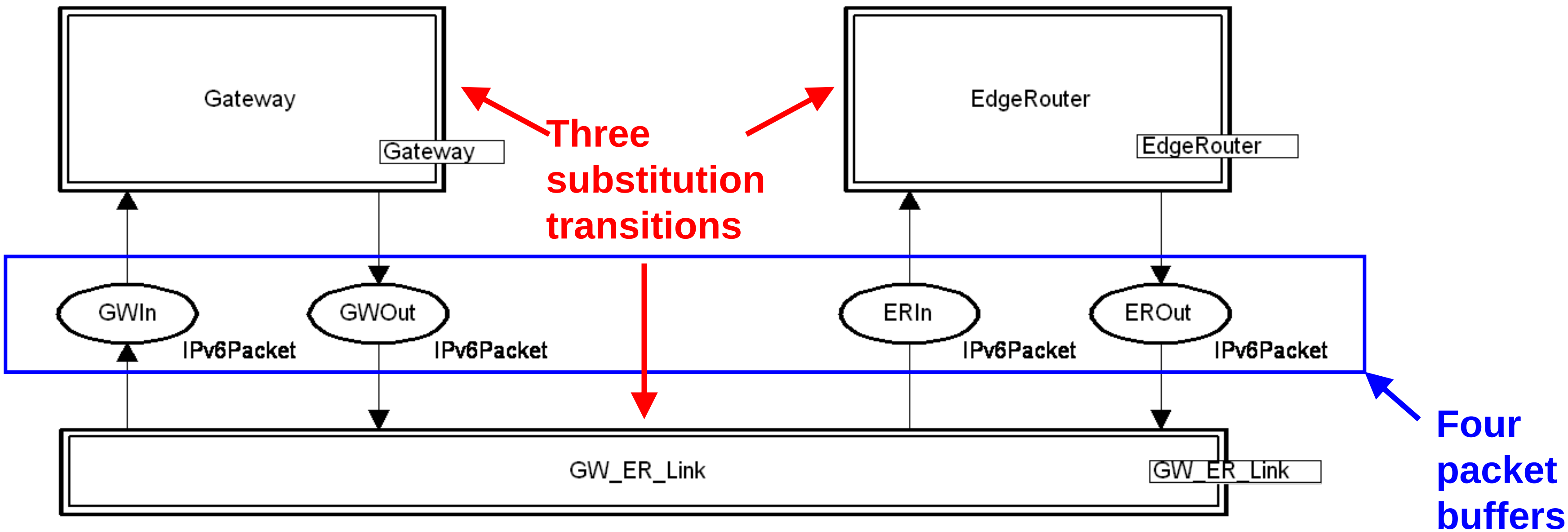
The Modelling Phase

- **CPN modelling applied for specification of the protocol software design:**
  - First a conventional natural language specification was developed by the protocol software engineers.
  - Protocol engineers was given a 6-hour course on CPNs.
  - Next a CPN model reflecting the specification was developed.
- **The ERDP protocol and the CPN model was then developed in an iterative process:**
  - **CPN model discussed and reviewed in each iteration.**
  - **CPN model used as a basis for discussion of protocol design.**
  - **Interactive simulation used for detailed investigations of the protocol software.**

Module Hierarchy



ERDP Top-level Module



Results from Modelling

- Several software design issues and errors were identified in the modelling phase:

Category	Review 1	Review 2	Total
Incompleteness and ambiguity in the ERDP specification	3	6	9
Errors in the protocol	2	7	9
Simplifications of the protocol	2	0	2
Additions	4	0	4
Total	11	13	24

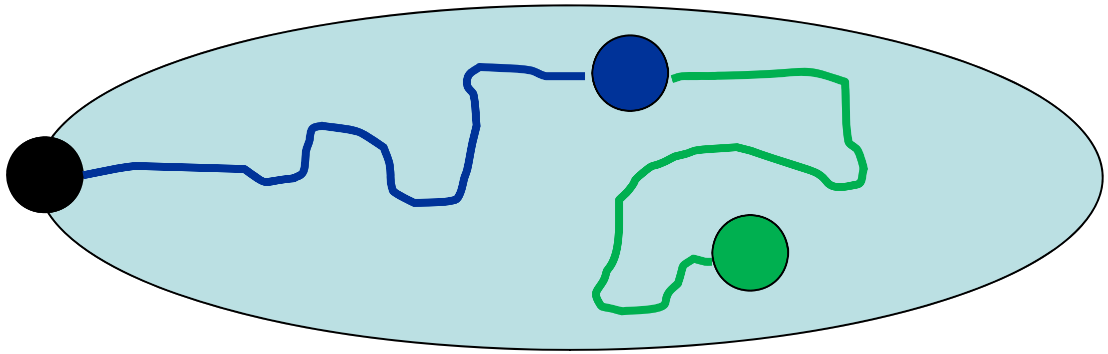
- Approximately 70 person-hours were used on CPN modelling and reviews.

State Space Exploration

- State space exploration was pursued after the three iterations of modelling.
- The first step was to obtain a finite state space:
  - The ERDP CPN model can have an arbitrary number of tokens on the packet buffers.
  - An upper integer bound of 1 was imposed on each of the packet buffers (GWIn, GWOut, ERIn, EROut).
  - This also prevents overtaking among the packets transmitted across the wireless link.
  - The number of tokens simultaneously on the four packet buffers was limited to 2.

Verification of ERDP

- **Key property of the ERDP protocol:**



From any state with a non-configured prefix P it is possible to reach a state where P is consistently configured.

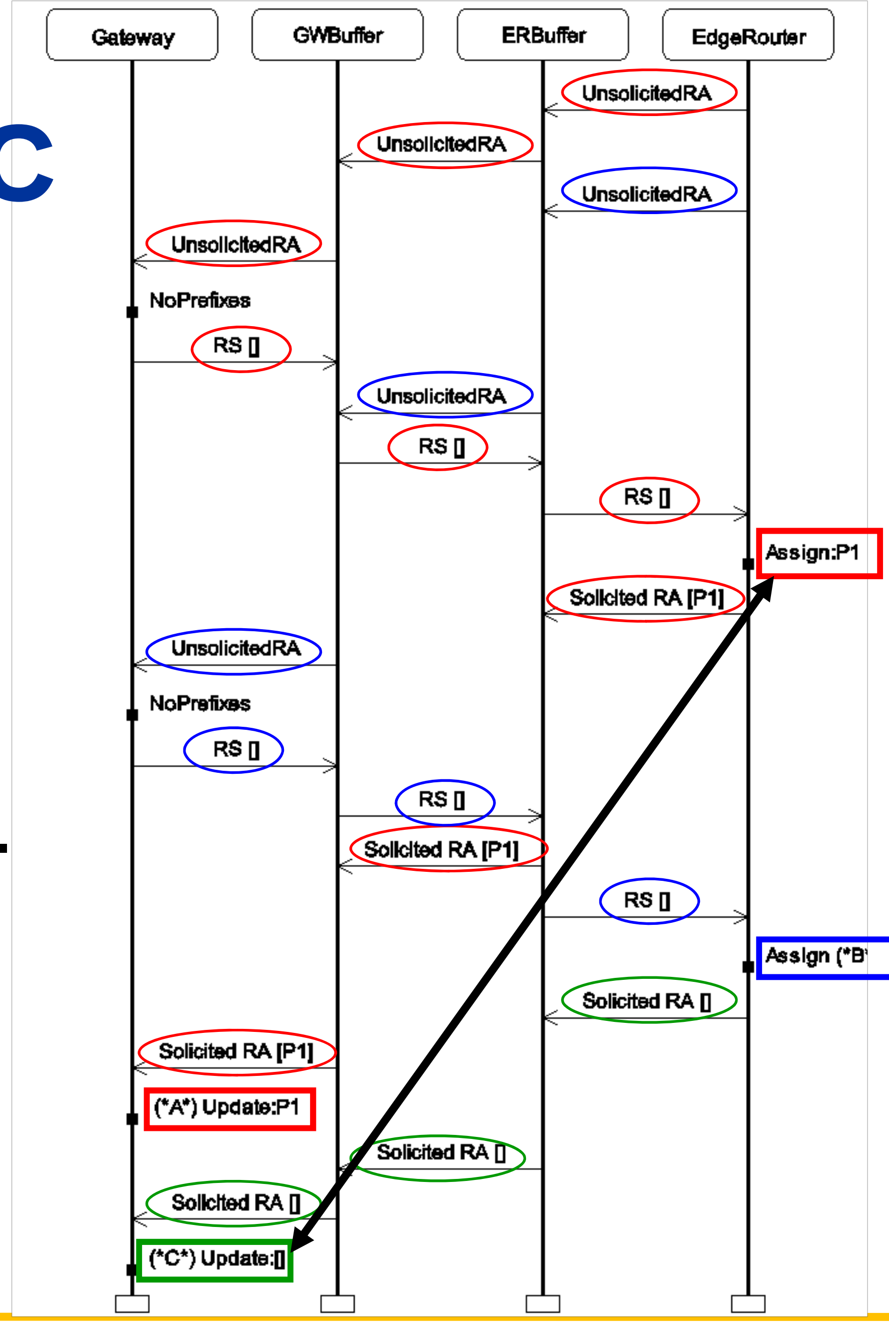
- **Investigated using state space exploration starting from the simplest possible configuration.**

Simplest Configuration
[one prefix, no loss, no expiration]

- **State space:** **46 nodes and 65 arcs.**
- **A single dead marking.**
- **Visual inspection showed that the dead marking is an inconsistently configured state:**
  - The edge router has assigned a prefix to the gateway.
  - BUT, the gateway is not configured with the prefix.
- **The error-trace was visualised by means of a message sequence chart.**
- **Demonstrates that errors tend to manifest themselves even in simple configurations.**

Error trace MSC

- **The edge router sends two unsolicited RAs.**
- **The first one gets through and we obtain a consistent configuration with prefix P1.**
- **When the second reaches the edge router there are no unassigned prefixes available.**
- **A Solicited RA with the empty list of prefixes is sent.**
- **The gateway updates its prefixes to be the empty list.**



Revised configuration
[One prefix, no loss, no expiration]

- **The protocol was revised such that the edge router always replies with the list of all currently assigned prefixes.**

- **State space: 34 nodes and 49 arcs.**
- **No dead markings and 11 home markings (constituting a single terminal SCC).**

- **Inspection showed that all home markings are consistently configured with the prefix.**
  - It is always possible to reach a consistently configured state for the prefix.
  - When such a state has been reached, the protocol entities will remain consistently configured.

Results from Verification

- **The verification was conducted in three steps where assumptions were gradually removed.**

- **Step 1 [no packet loss and no expire of prefixes]:**
  - Synchronisation error between edge router and gateway.
  - The error was corrected and the key property was verified.
- **Step 2 [packet loss on wireless link added]:**
  - Synchronisation error when certain unsolicited RAs was lost.
  - Livelock error in processing of router advertisement in gateway.
  - The errors were corrected and the key property was verified.
- **Step 3 [expire of prefixes added]:**
  - Property verified: Consistent configuration always possible.

State Space Statistics

|P|	No loss/No expire		Loss/No expire		Loss/Expire	
1	34	49	68	160	173	531
2	72	121	172	425	714	2,404
3	110	193	337	851	2,147	7,562
4	148	265	582	1,489	5,390	19,516
5	186	337	926	2,390	11,907	43,976
6	224	409	1,388	3,605	23,905	89,654
7	262	481	1,987	5,185	44,450	169,169
8	300	553	2,742	7,181	78,211	300,072
9	338	625	3,672	9,644	130,732	505,992
10	376	697	4,796	12,625	209,732	817,903

- **When a state space had been generated, the verification of the key properties was be done in a few seconds.**

Lessons Learned

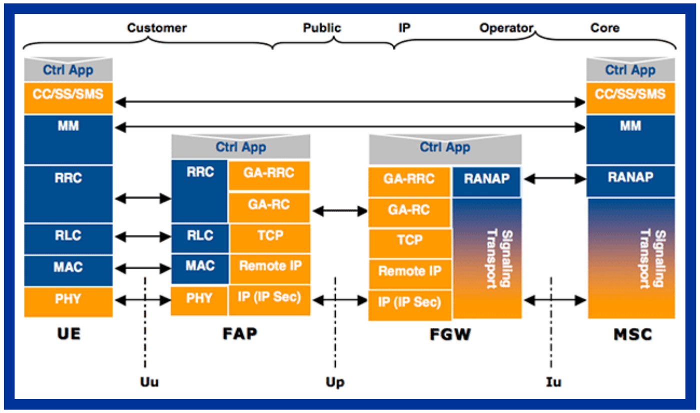
- **Start state space exploration from the simplest possible configurations:**
  - Errors often manifest themselves in the simplest configurations and with the strongest assumptions.
  - The assumptions are then gradually lifted and larger configurations considered.
- **For the ERDP protocol state explosion was not a problem.**
- **The key properties could be verified for the number of prefixes envisioned in practice.**
- **Both modelling and state space exploration played a central role in validating the protocol.**

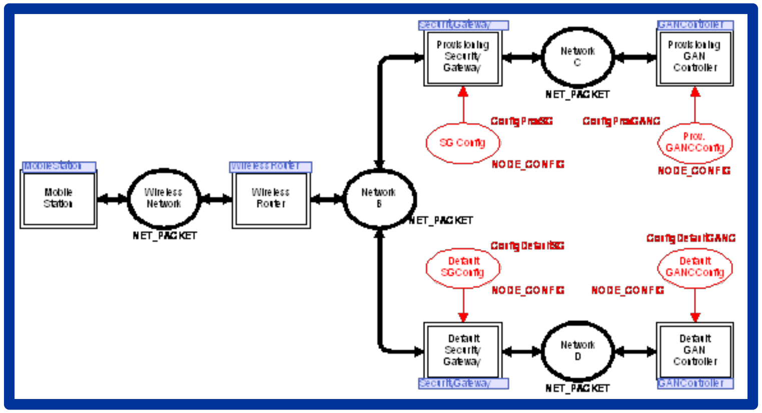
Conclusions from Project

- The construction the CPN model improved the quality of the ERDP design specification.
- Non-trivial design errors were identified and fixed in the course of modelling and verification.

- CPN and CPN Tools were powerful enough to specify and validate real-world protocol software.
- Approximately 100 person-hours over 4-months were used for modelling and verification.

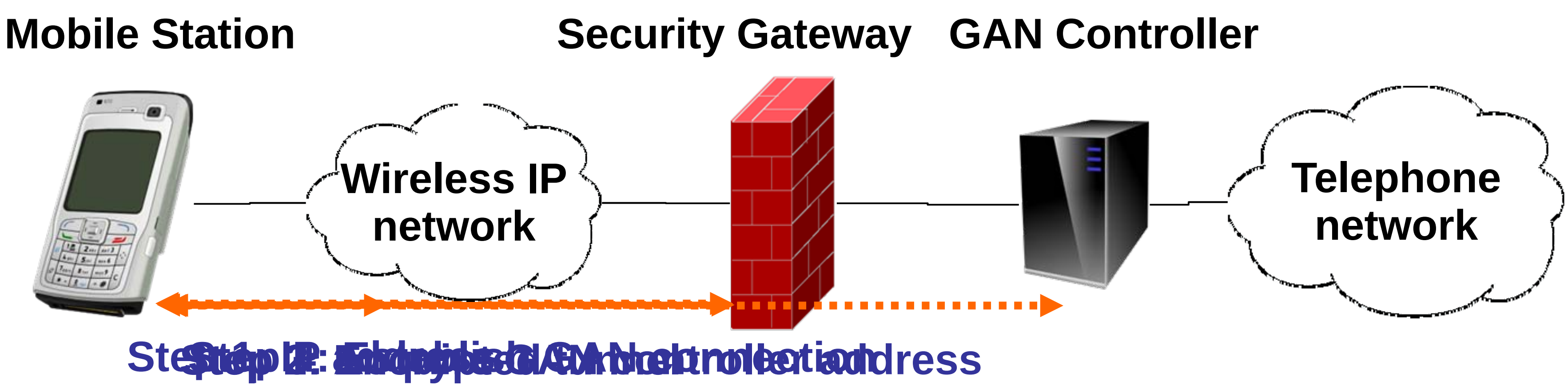
Formal Specification and Validation of Secure Connection Establishment in a Generic Access Network Scenario





The GAN Architecture

- This subproject is concerned with the **Generic Access Network** (GAN) architecture.
- Currently being developed by the 3rd Generation Partnership Project [www.3gpp.org].
- Supports access to telephone network services (e.g., messaging and voice calls) via IP networks:



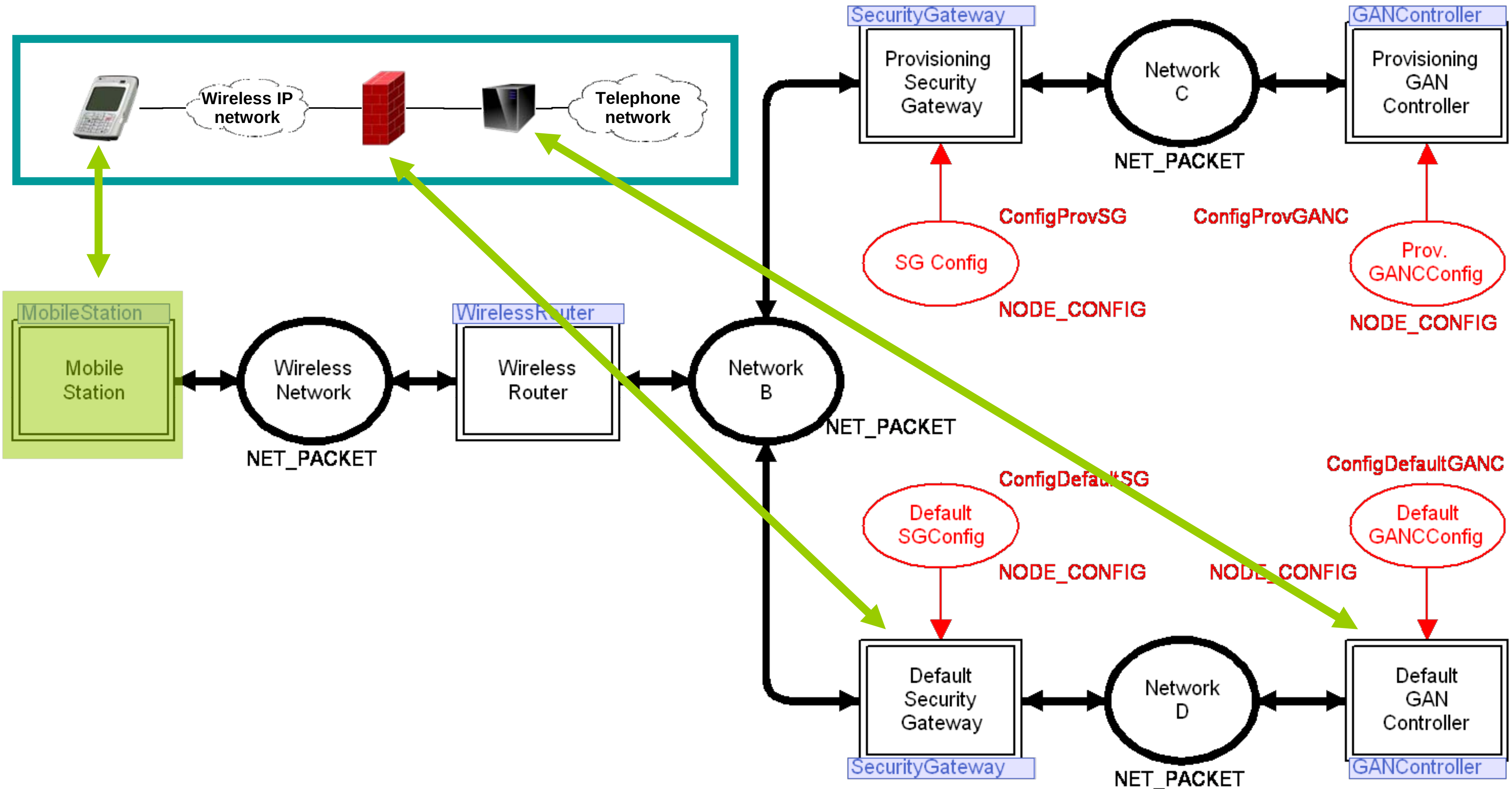
GAN at TietoEnator

- **A specific instantiation of the GAN architecture:**
  - Define the scope of the protocol software to be developed.
  - Specify detailed design and usage of the protocol software.
- **Main purpose of the modelling was to specify the use of:**
  - The Dynamic Host Configuration Protocol (DHCP) for IP address configuration of the mobile station.
  - The IP security (IPsec) protocols for encryption and authentication.
  - The use of the Internet Key Exhange (IKE) protocol for negotiation of IPsec parameters.
- **Use simulation and state space analysis to validate the completeness and correctness of the GAN scenario.**

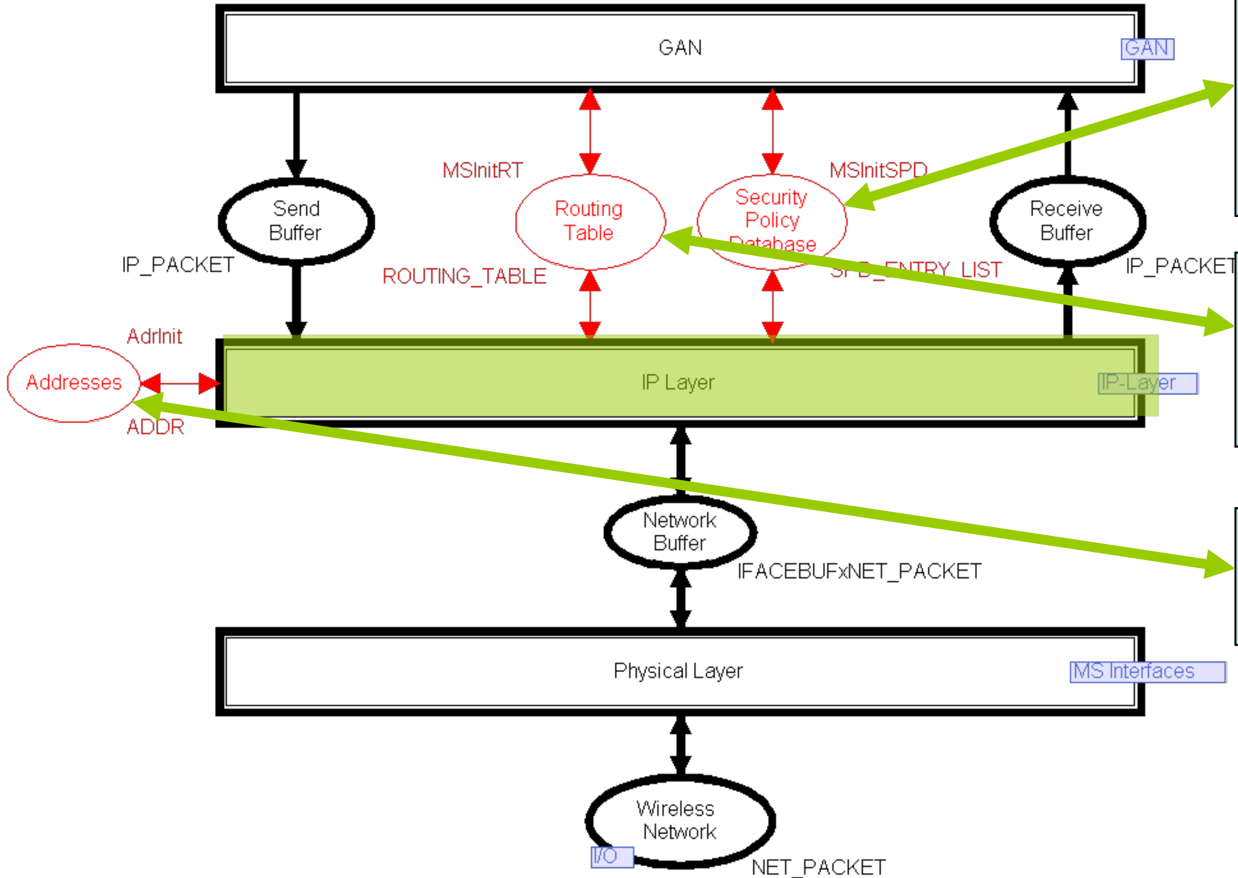
CPN Model Overview

- **A hierarchical CPN model consisting of 31 modules organised in four levels:**
  - Network nodes: the mobile station, wireless router, security gateway(s), and the GAN controller(s).
  - Protocol entities: DHCP, IPsec, IKEv2, GAN signalling, and the Internet Protocol (IP) network layer.
- **Developed in close interaction with TietoEnator protocol engineers over a period of 3 months.**
- **Initial CPN model constructed based on a textual description of the GAN Scenario considered.**
- **Protocols engineers did not have any previous knowledge of Coloured Petri Nets.**

Top-level Module: Network Nodes



Mobile Station



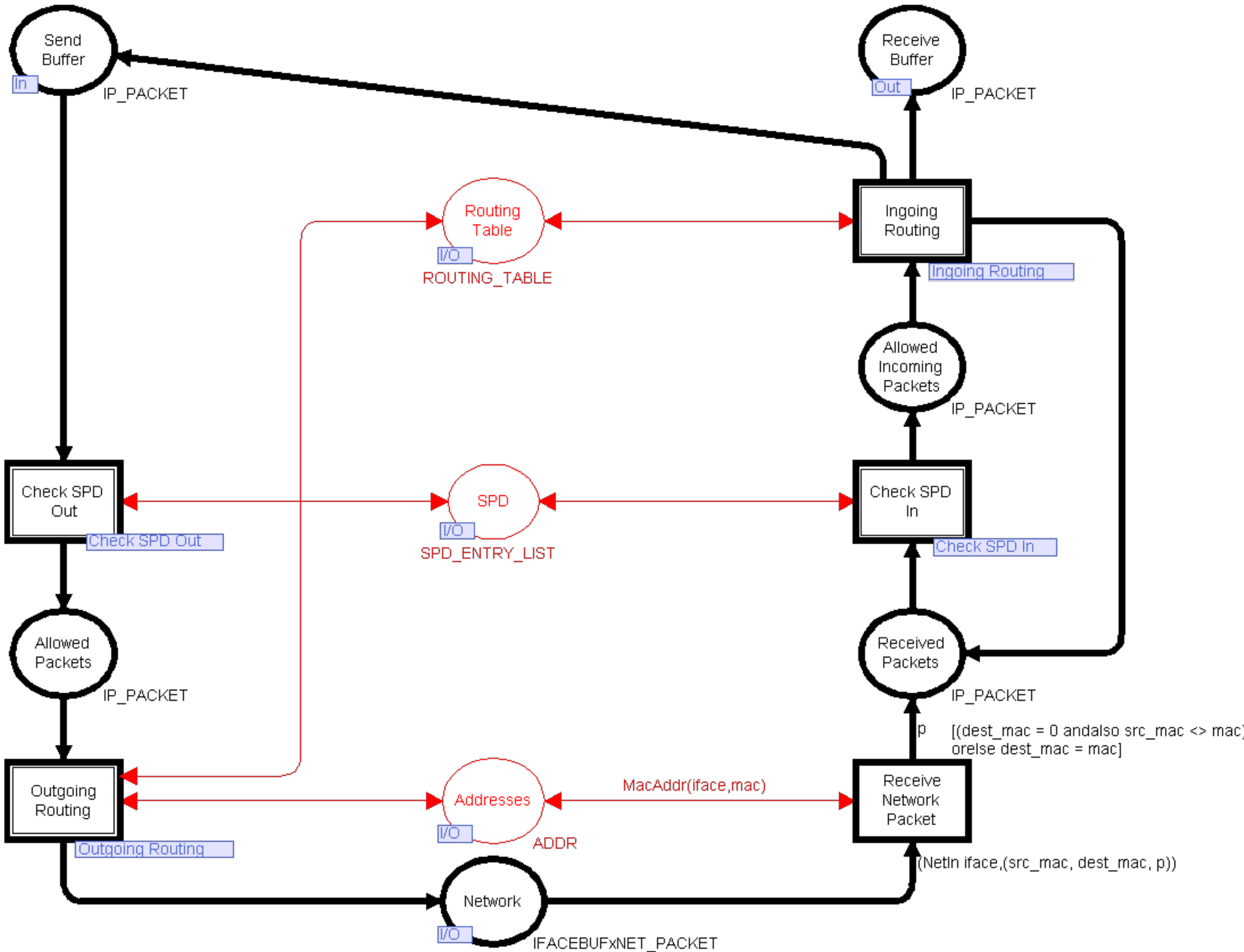
SPD: used by IPsec to specify which packets are allowed to be sent/received by the IP layer module

RT: used in conjunction with encapsulation/decapsulation and the secure tunnels.

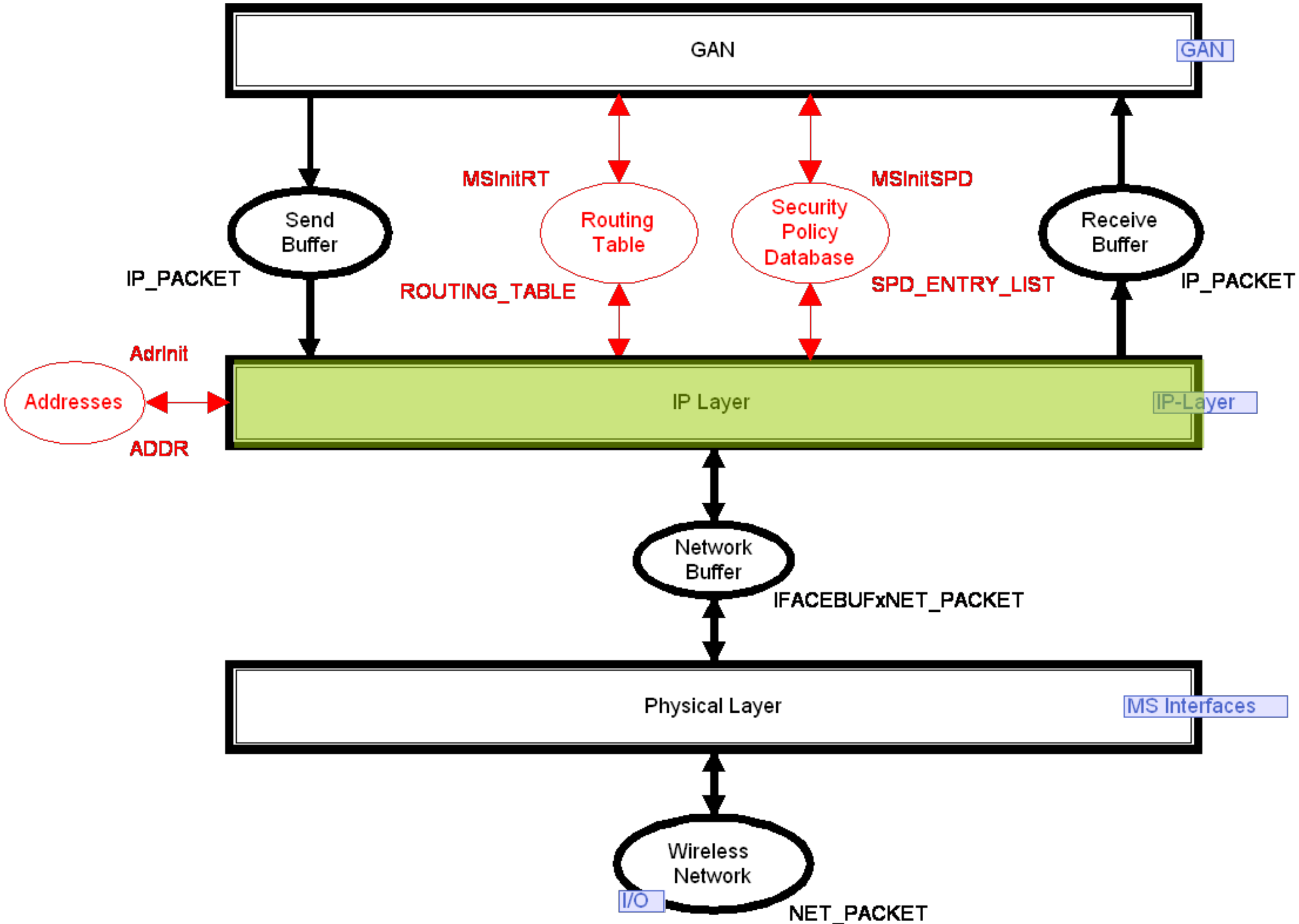
Adr: used for configuration of the IP layer module.

- All network nodes structured similarly and reuses the IP and physical layer modules.

IP Layer Modelling

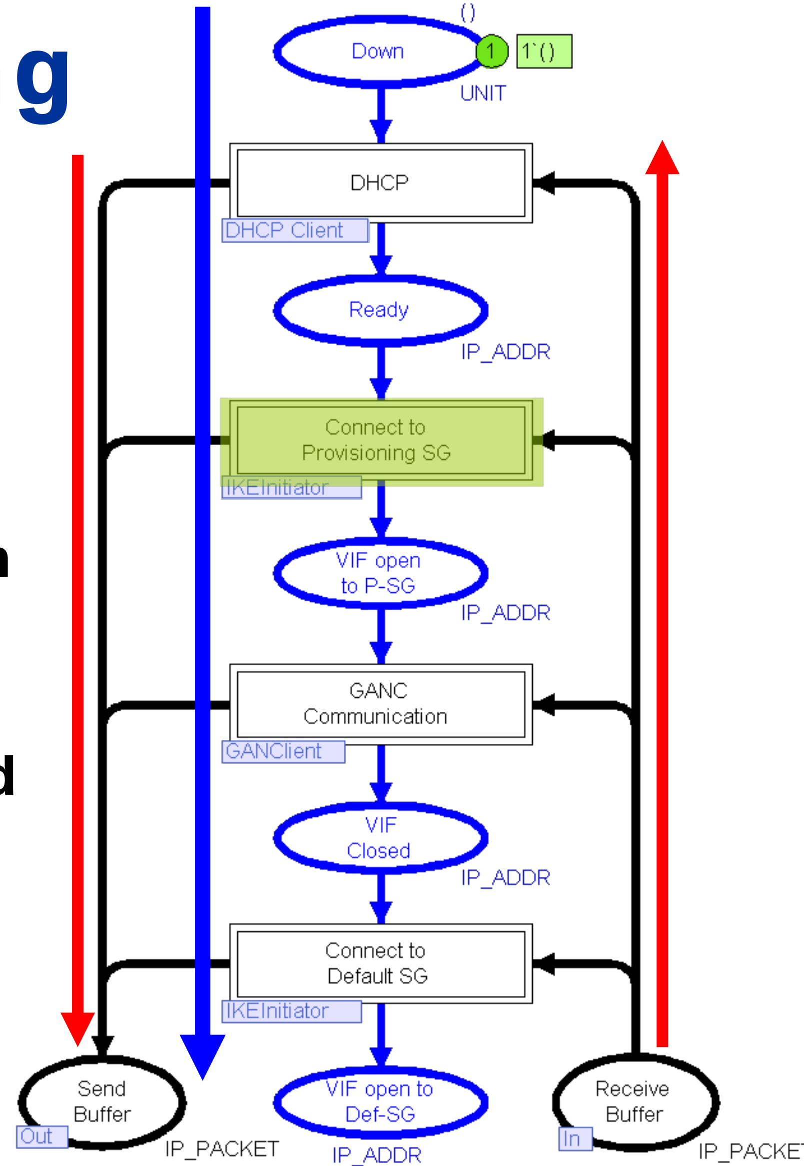


Mobile Station



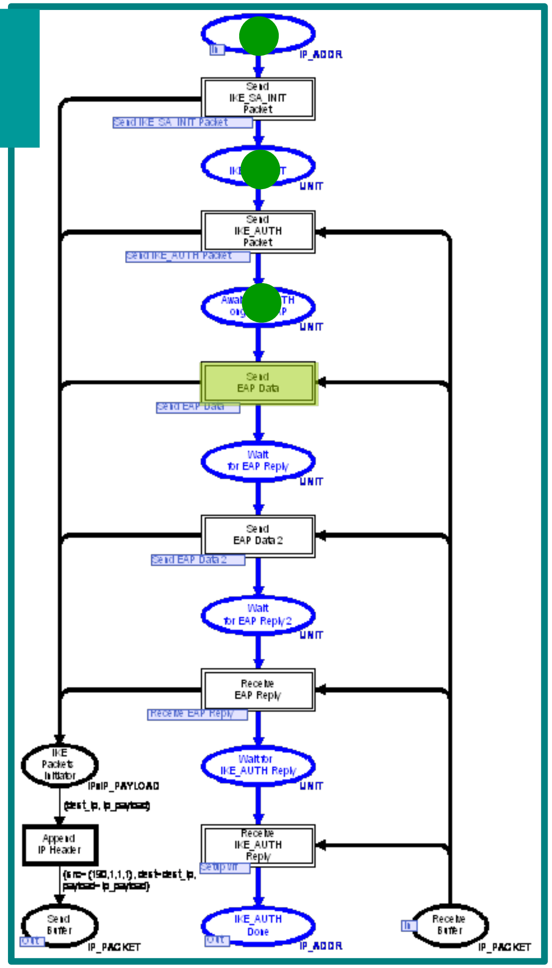
GAN Layer Modelling [Mobile Station]

- Specifies the steps in establishing a GAN connection.
- Explicit graphical representation of control flow and packet flow.
- The security policy database and routing table are accessed and updated in the individual steps.
- GAN layer of other network nodes are organised similarly.

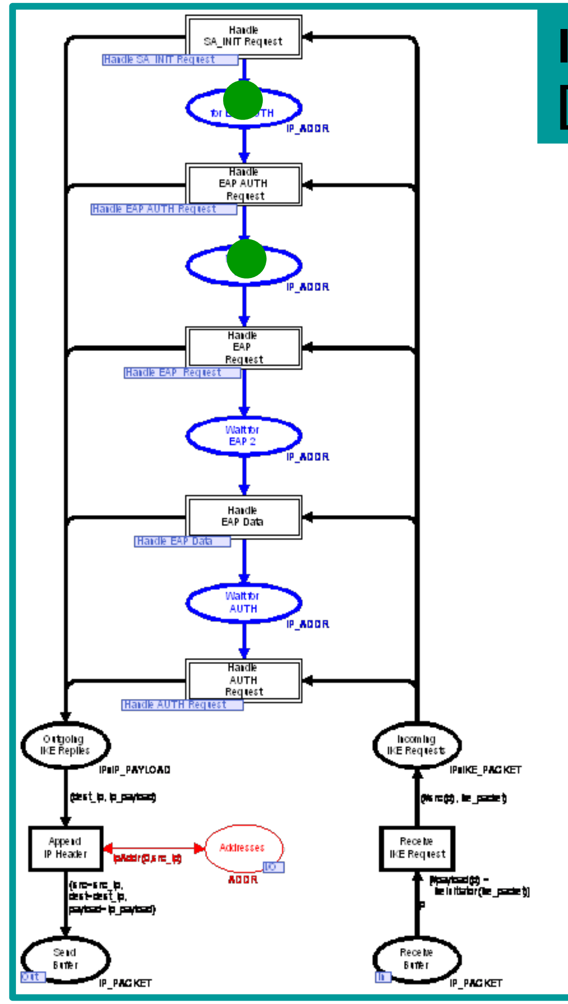


Modelling Protocol Entities

IKE initiator [mobile station]



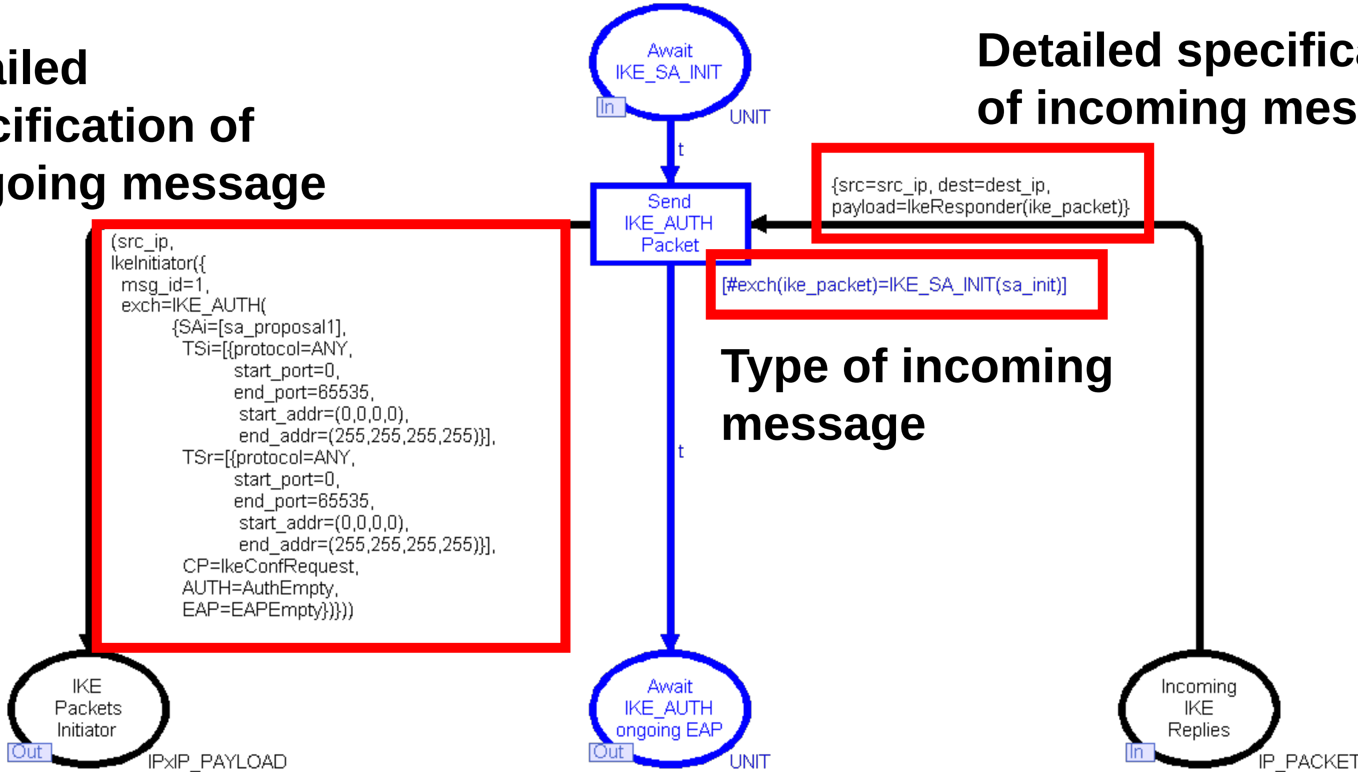
IKE responder [security gateway]



Modelling Protocol Entities

Detailed specification of outgoing message

Detailed specification of incoming message



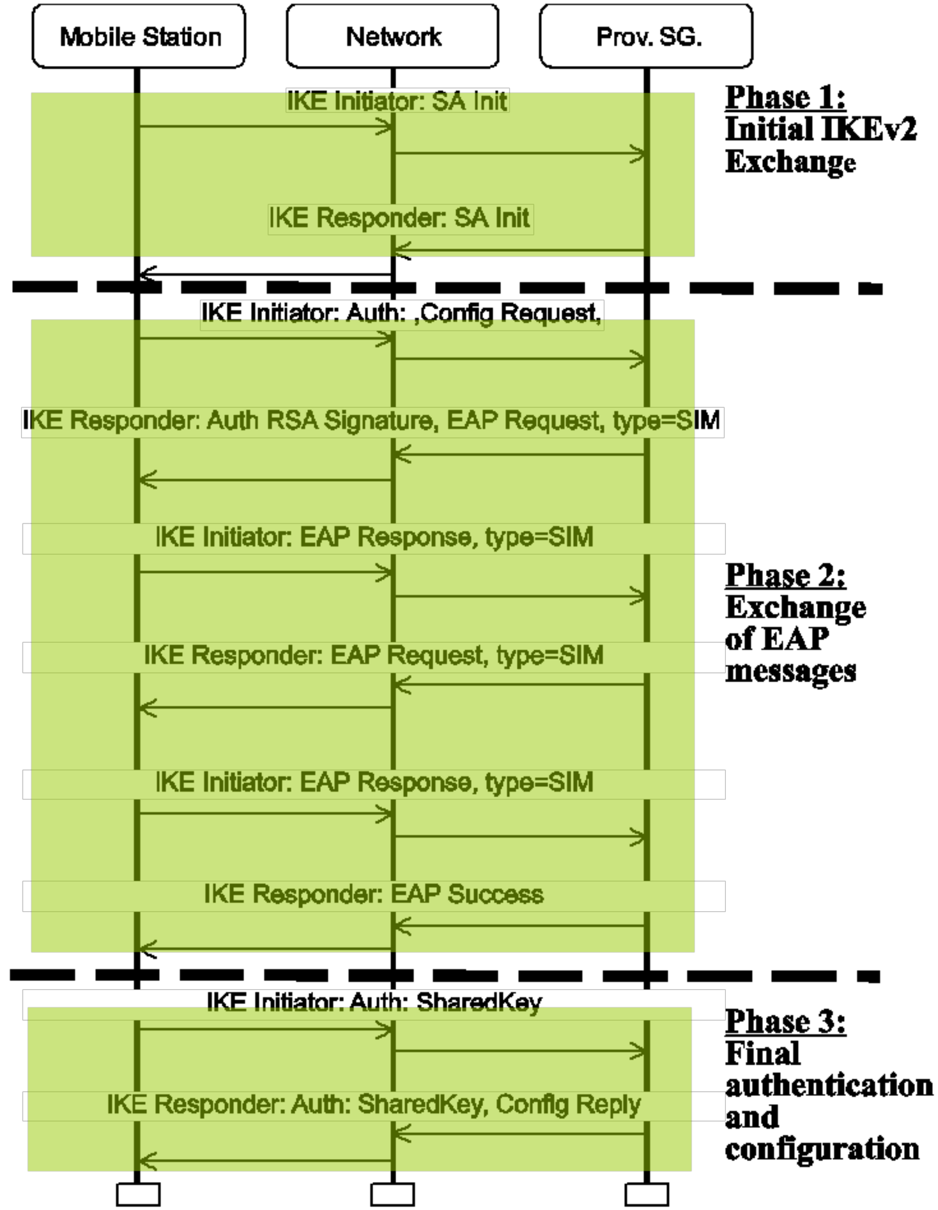
- **The modelling level at which building an executable model was powerful in making design issues explicit.**

Simulation and MSCs

- Single-step simulation and message sequence charts (MSCs) were used for initial validation.
- Detailed inspection of control flow, packet flow, security policy databases, and routing tables.
- Conducted jointly with the protocol engineers at TietoEnator in two formal meetings.
- Enabled discussions and resulted in several further modifications to the CPN model.

Example MSC

- IKE phases of step 2 in GAN connection establishment.
- Generated directly from the CPN model using the BRITNeY animation tool.
- Presents the operation of the protocols in a form well-known to protocol engineers.
- Focus on message exchange between peer protocol entities.



State Space Analysis

- State space analysis was subsequently used to verify the GAN connection establishment.
- Key correctness criteria:

> *Always possible to reach a state where the GAN connection is properly established [AG EF φ]*

- State space has 3,854 nodes and a single dead marking M which is also a home marking.
- M represents a state where the connection has been properly established.

Conclusions

- **The construction of the CPN model:**
  - Used to specify the specific instance of the GAN architecture to be implemented by TietoEnator.
  - Developed and reviewed in close co-operation with TietoEnator protocol engineers.
  - Spans multiple protocols and protocol layers which is a key characteristic of the GAN architecture.
- **Benefits of the CPN model for development:**
  - Useful in capturing the scope and initial design of the protocol software to be developed.
  - Useful in detailing and validating the message exchanges that were not explicit in the initial textual GAN specification.
  - A high degree of confidence in the design has been obtained.

Verification in Perspective

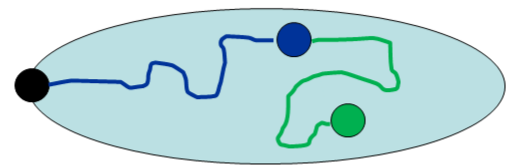
- **Modelling and state space verification for system validation goes hand in hand:**

Ericsson Edge Router Discovery Protocol Project

Modelling Phase

Category	Review 1	Review 2	Total
Incompleteness and ambiguity in the ERDP specification	3	6	9
Errors in the protocol	2	7	9
Simplifications of the protocol	2	0	2
Additions	4	0	4
Total	11	13	24

Verification Phase



Three subtle behavioural errors found

- **When pragmatically applied current methods can be used to obtain useful results on real systems:**
  - **Compact CPN modelling means that the full state space can usually be explored for the smallest configurations.**
  - **Advanced methods in many cases allow the system configurations occurring in practice to be verified.**