# LMAP Framework

## draft-eardley-lmap-framework-02

Philip Eardley

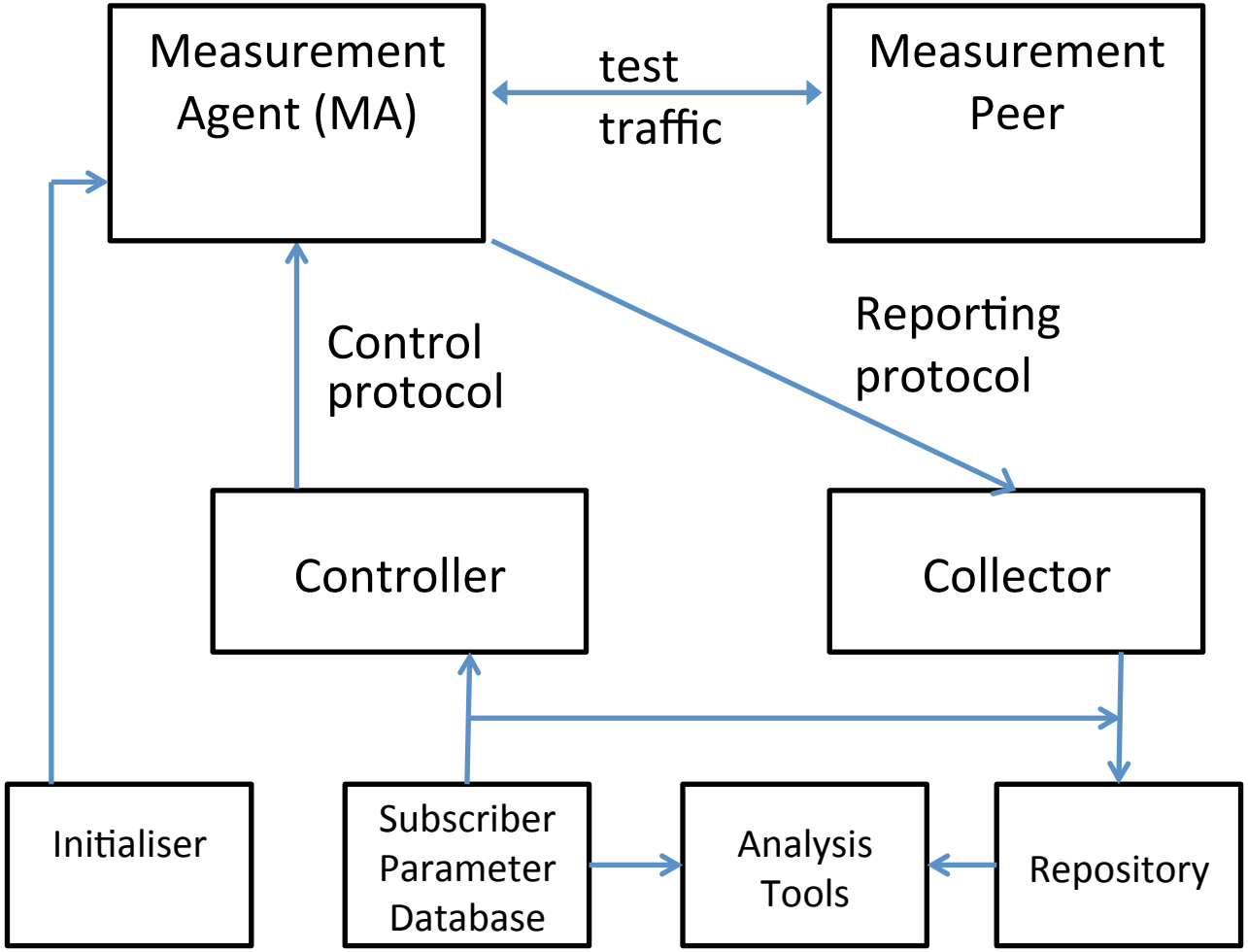
30th July 2013

Berlin, IETF-87

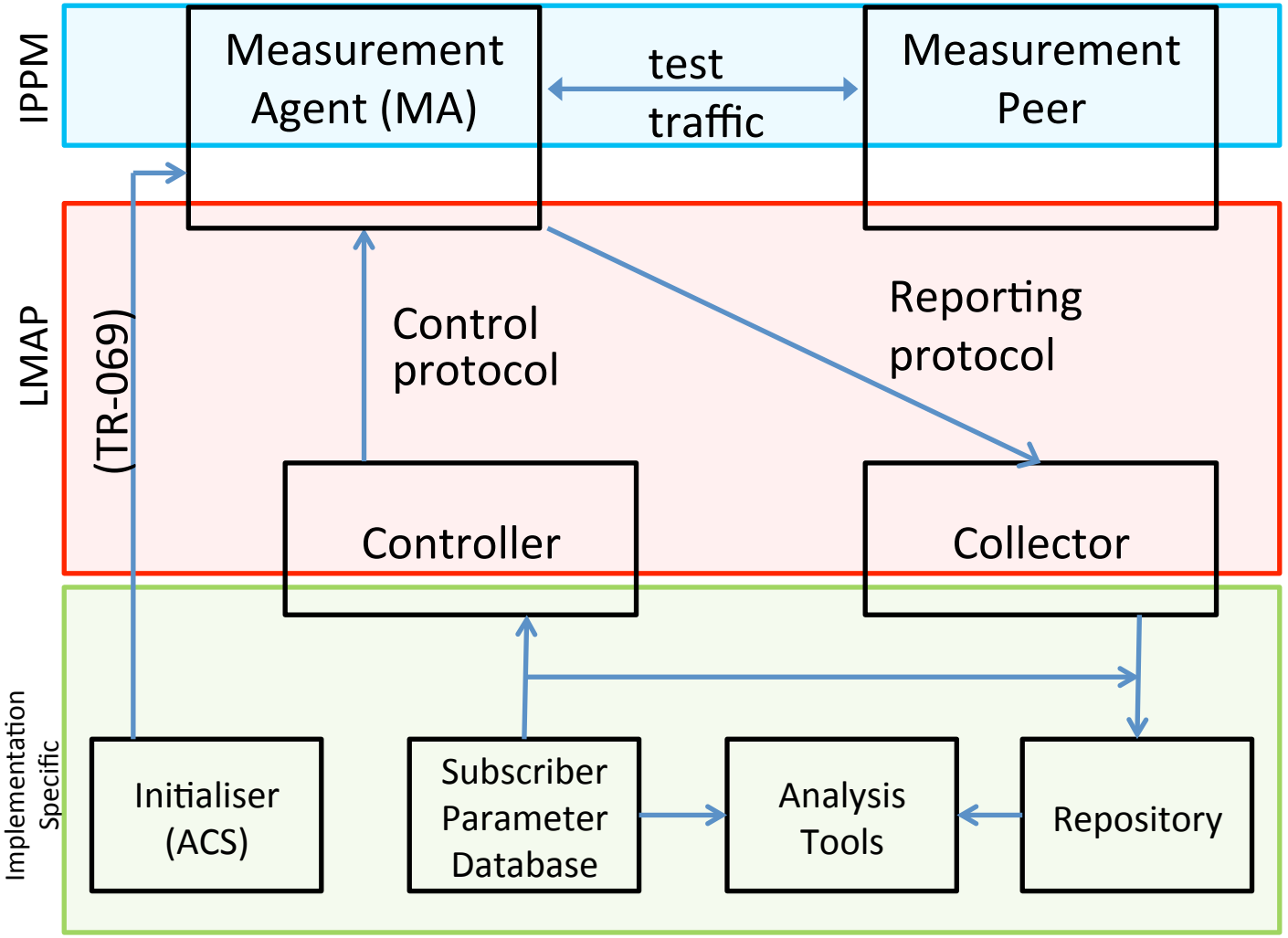
# Aim

- Towards our first milestone on the Framework (Sept 2013, WG I-D)
- It could also be a place to collect open issues of an architectural nature
  - Current i-d starts to do this

# Basic architectural elements



# Basic architectural elements




- Tests
- Registry
- Path Definition
- Framework
- Information model
- Data model
- Control Protocol
- Reporting Protocol
- Initialisation
- TR-069 based data model & protocols
- Characterisation plan

# Constraint #1: Measurement system is under the direction of a single organisation

- Single organisation responsible for both data and user experience
  - Simplifies solution as avoids policy decisions and coordination
  - (but the deployed components of a single measurement system may span ownership & admin boundaries)
- Inter-organisation coordination is potential topic after future re-chartering
  - (for both control & collection)
  - Interesting but raises new issues
  - Out of scope at this stage

# Constraint #2: Each Measurement Agent has only a single Controller at any point in time

- Single Controller determines MA's Schedule
  - So MA does not have to manage contention between multiple, conflicting Schedules
  - Simplifies MA design and deployment
- Note, an operator may have several Controllers
  - For different device types, scalability, resilience etc

# Constraint #3: A Measurement Agent acts autonomously

- MA operates tests and reports results without further reference to Controller (once it gets Schedule)
  - Avoids frequent checks with Controller
  - MA (on edge /end device) knows when not to run test due to user activity

# Constraint #4: the WG doesn't consider 'gaming the system'

- 'gaming the system': in theory an operator could prioritise traffic on the lines that regulator monitored
- Consideration is out of scope
  - The issue can be solved in non-technical ways (eg a code of conduct…)

# Constraint #5: Measurement Agent is most likely behind a NAT

- So MA will pull its Instruction from the Controller

# Merging with the 2 framework drafts

- Charter says doc "provides common terminology, basic architecture elements, and justifies the simplifying constraints"
- Proposed starting point:
- Intro – set the context – TBD
- Terminology – S3 of draft-eardley-lmap-terminology
- Basic architecture elements
  - S3 of draft-akhter-lmap-framework for the 4 basic functions
  - New text to outline the interactions of these 4 functions
  - Also briefly describe the other elements beyond WG's scope
- Simplifying constraints – S3 of draft-eardley-lmap-framework
  - And similar issues (deployment considerations)

# How to handle ‘open issues’?

- Should the Framework i-d document open issues and their resolution?
  - Probably yes, as these would form starting points for the Informational Model & Protocol work
- Open issues:
  1. Should there be negotiation between a Controller and its MA, or should the Controller simply instruct the MA by sending its Test and Report Schedules?
  2. Please suggest architectural issues we need to resolve!