PM Data Collection and Storage

Front-end of the post-mortem system

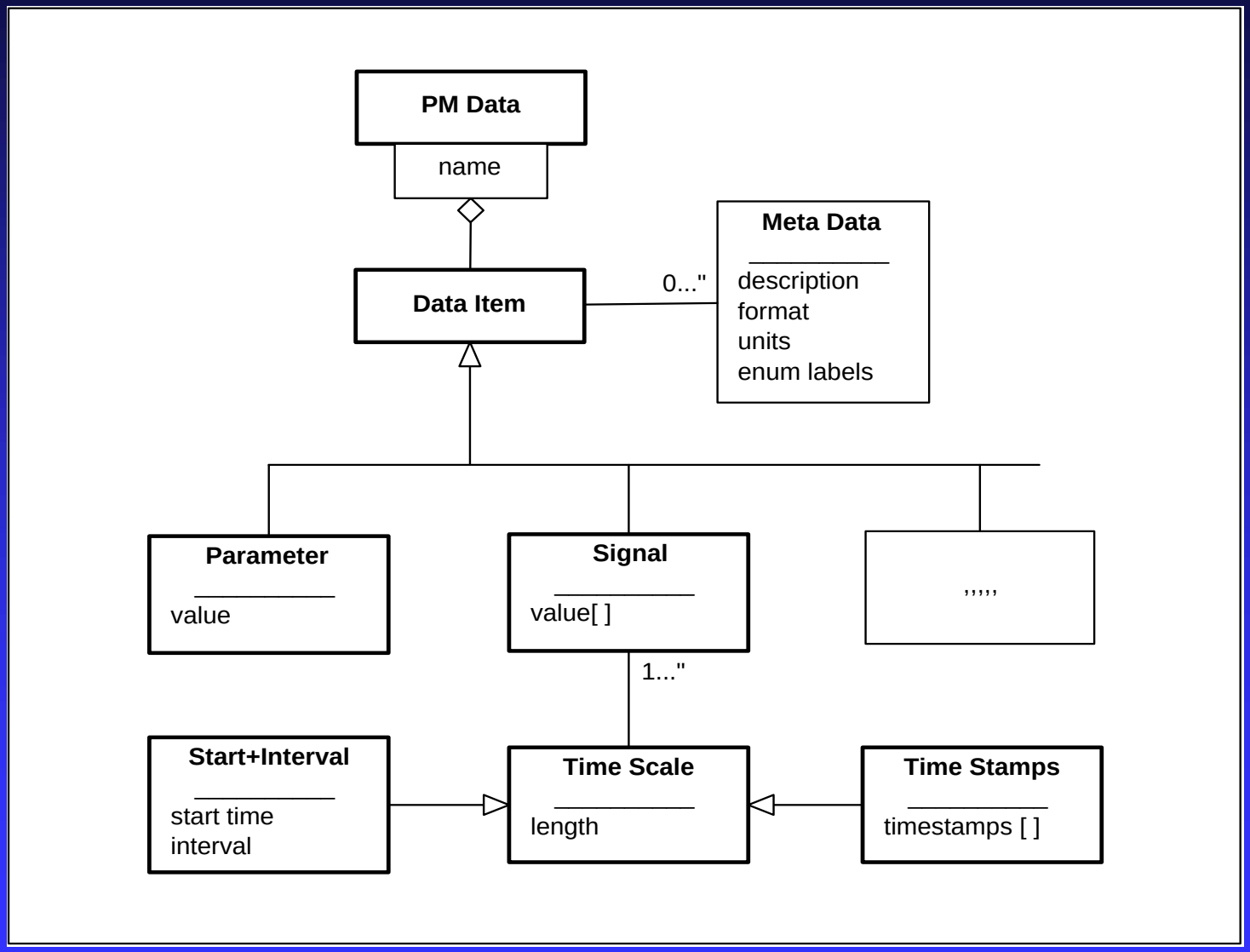
User interface

The interface that clients use to export data to the PM system.

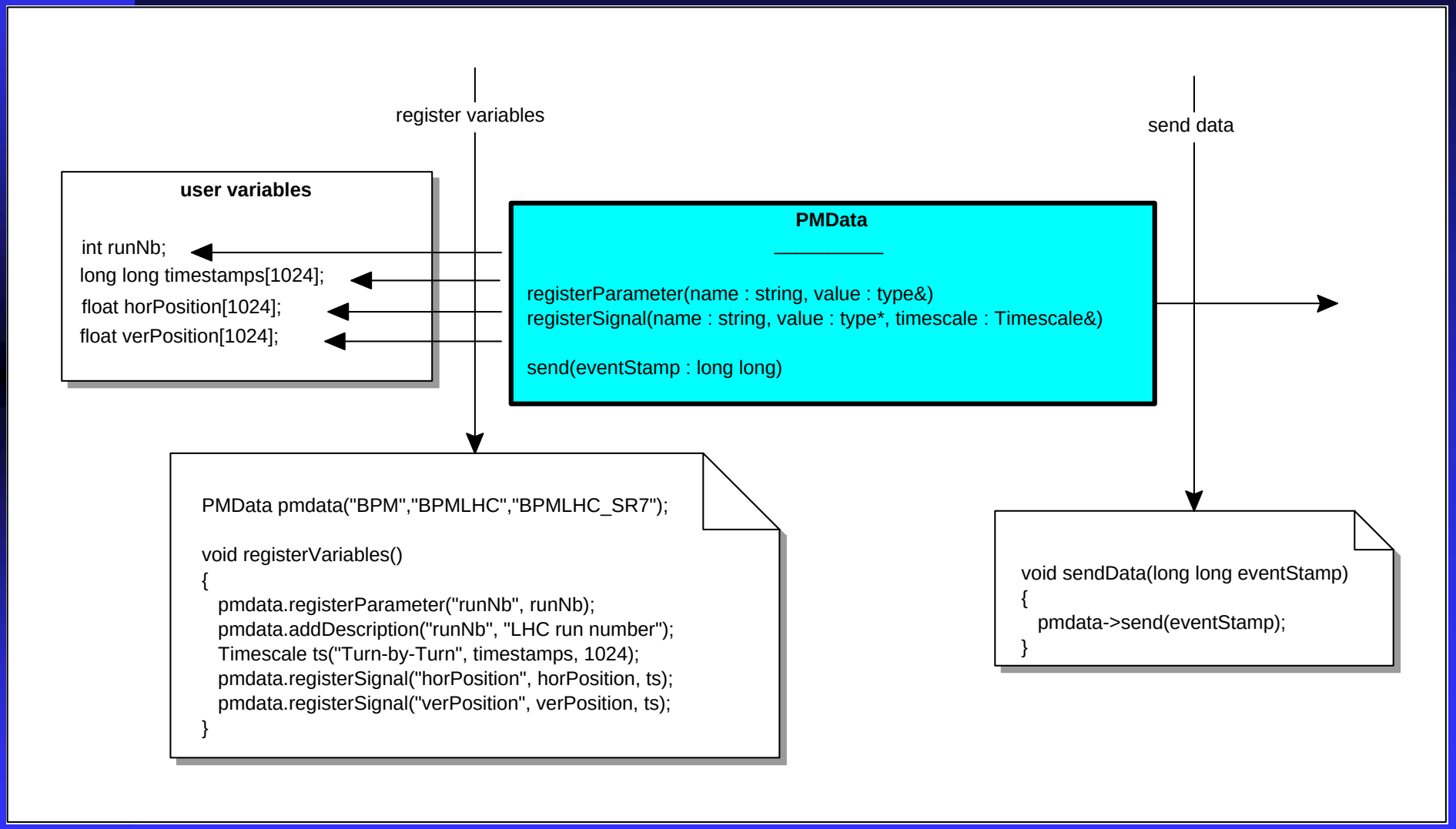
Two approaches to data formatting and transfer:

- "Memory dump"
  - Memory image of the user data is sent to the PM system as a byte array and decoded on the data processing side using some form of format conversion.
  - Very flexible and easy to use on the client side, but
  - requires specific decoding software and the data conversion is rather slow.
- "Data objects"
  - User data is mapped to an object, a binary (CORBA CDR) image of the object is stored in the PM system.
  - Objects can be reconstructed and directly used in data processing applications, no need to write specific decoders.
  - A bit more work on the client side, but more fast and generic data processing software.

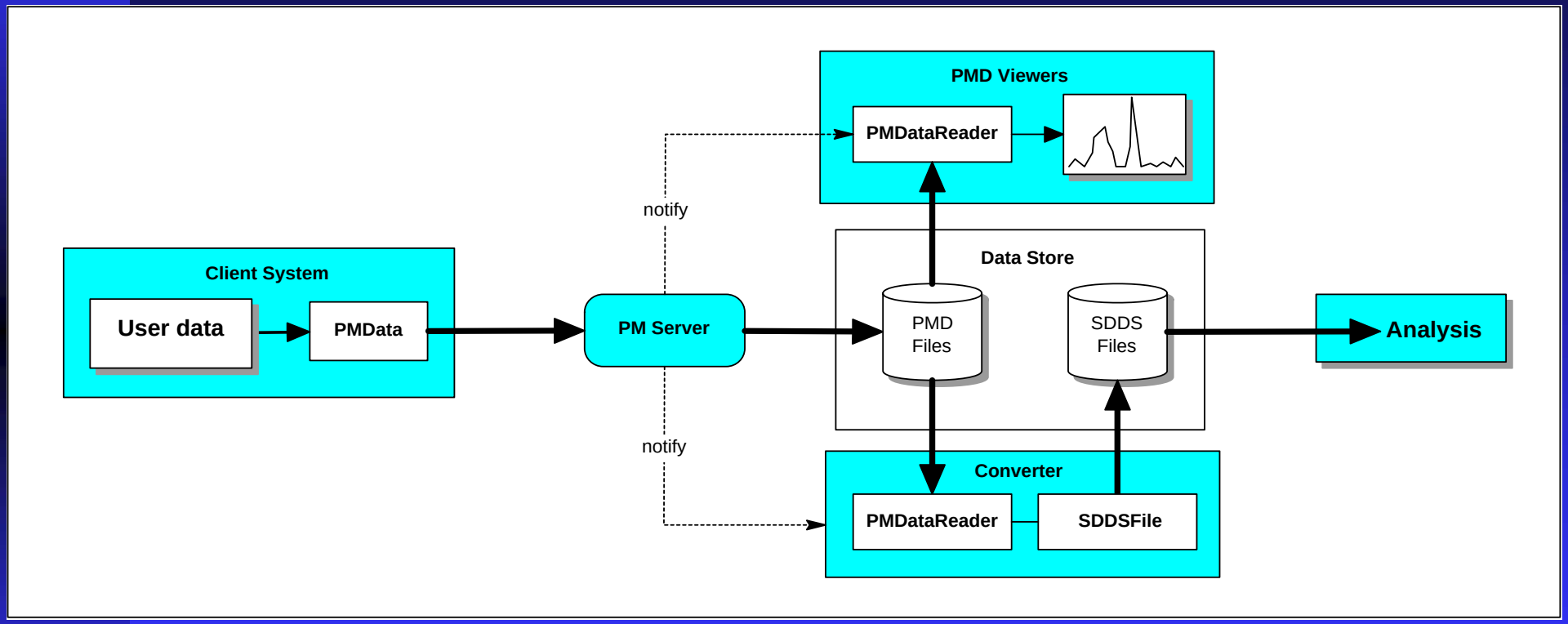
Data model


PM Data
name
Meta Data
description
format
units
enum labels
0..."
Data Item
Parameter
value
Signal
value[ ]
,,,,,
1..."
Start+Interval
start time
interval
Time Scale
length
Time Stamps
timestamps [ ]

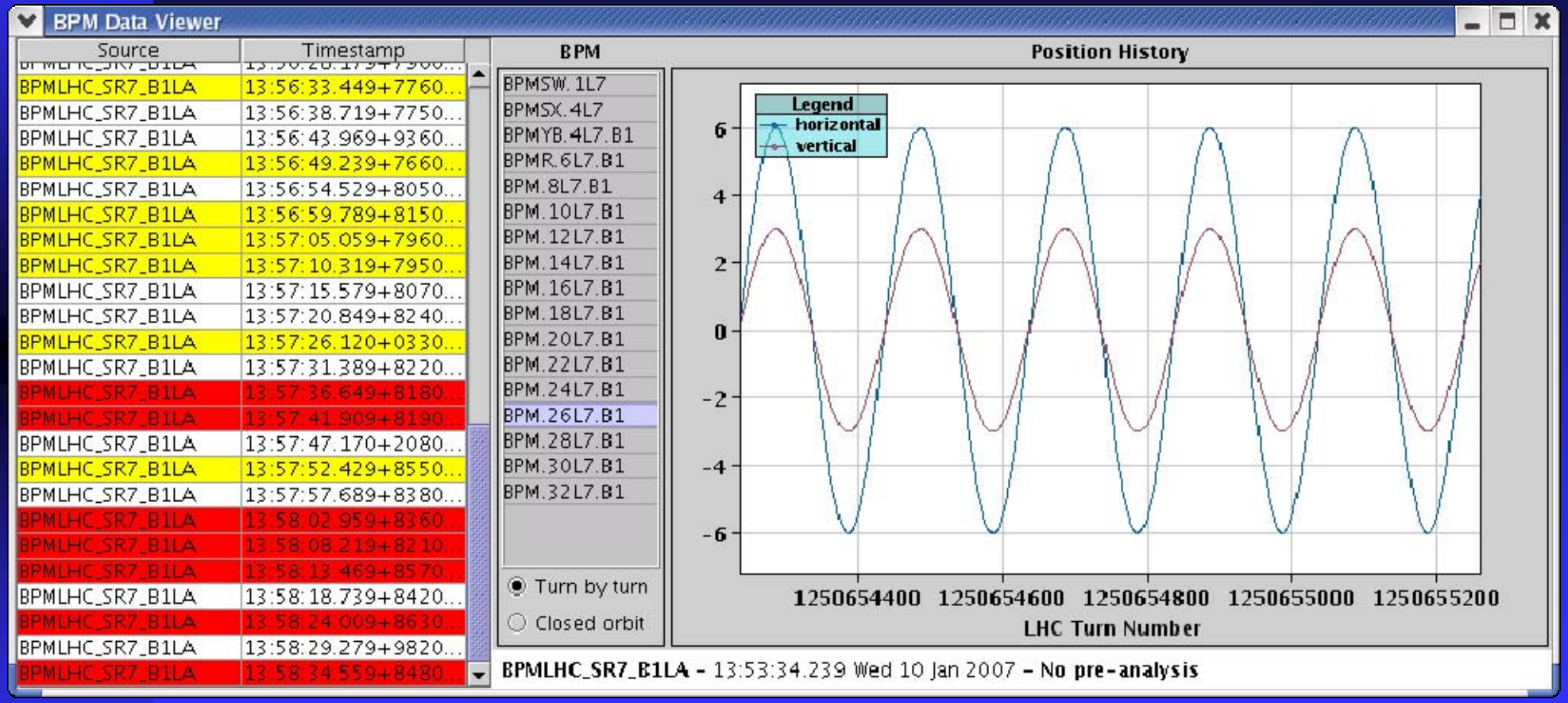
Client API


register variables
send data
user variables
int runNb;
long long timestamps[1024];
float horPosition[1024];
float verPosition[1024];
PMData
registerParameter(name : string, value : type&)
registerSignal(name : string, value : type*, timescale : Timescale&)
send(eventStamp : long long)
PMData pmdata("BPM","BPMLHC","BPMLHC_SR7");
void registerVariables()
{
pmdata.registerParameter("runNb", runNb);
pmdata.addDescription("runNb", "LHC run number");
Timescale ts("Turn-by-Turn", timestamps, 1024);
pmdata.registerSignal("horPosition", horPosition, ts);
pmdata.registerSignal("verPosition", verPosition, ts);
}
void sendData(long long eventStamp)
{
pmdata->send(eventStamp);
}

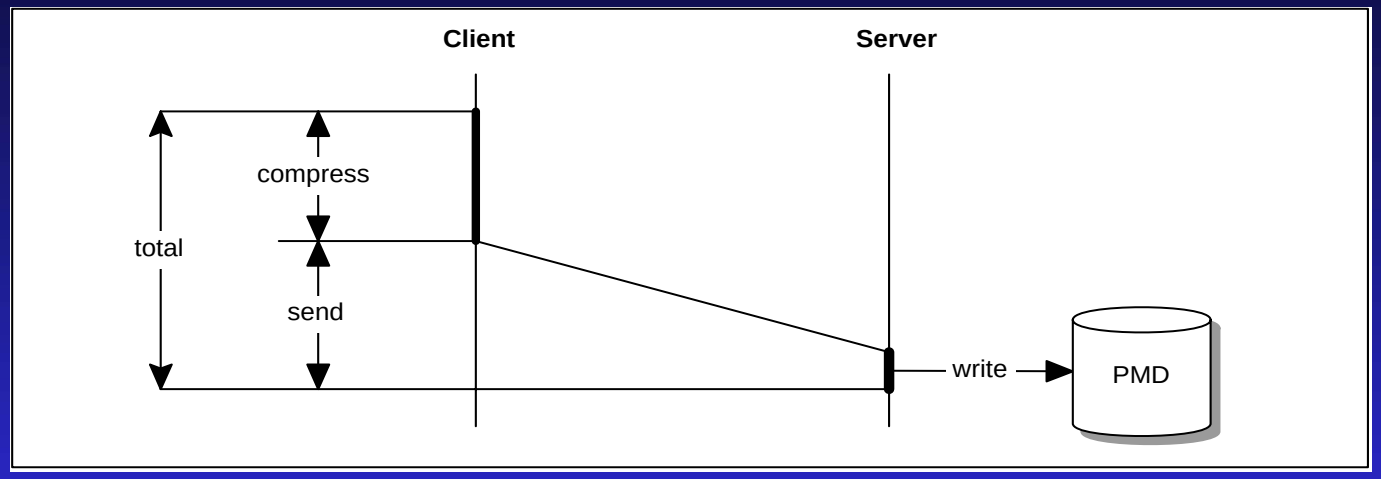
Data flow


PMD Viewers
PMDataReader
notify
Client System
User data
PMData
PM Server
Data Store
PMD
Files
SDDS
Files
Analysis
notify
Converter
PMDataReader
SDDSFile

BPM data viewer



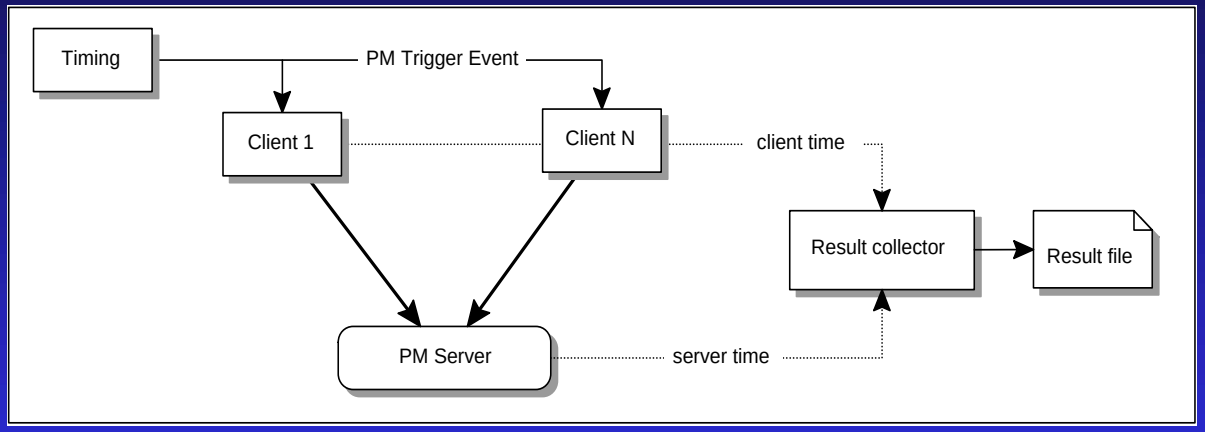
Performance



Source	Raw size	Compress time	Compressed size	Send time	Write time	Total time
QPS	520420	14 ms	29960	4 ms	0.4 ms	18 ms
FGC	1543536	117 ms	265527	29 ms	3.1 ms	146 ms

Scalability (1/4)

Scalability test framework by Joel Lauener

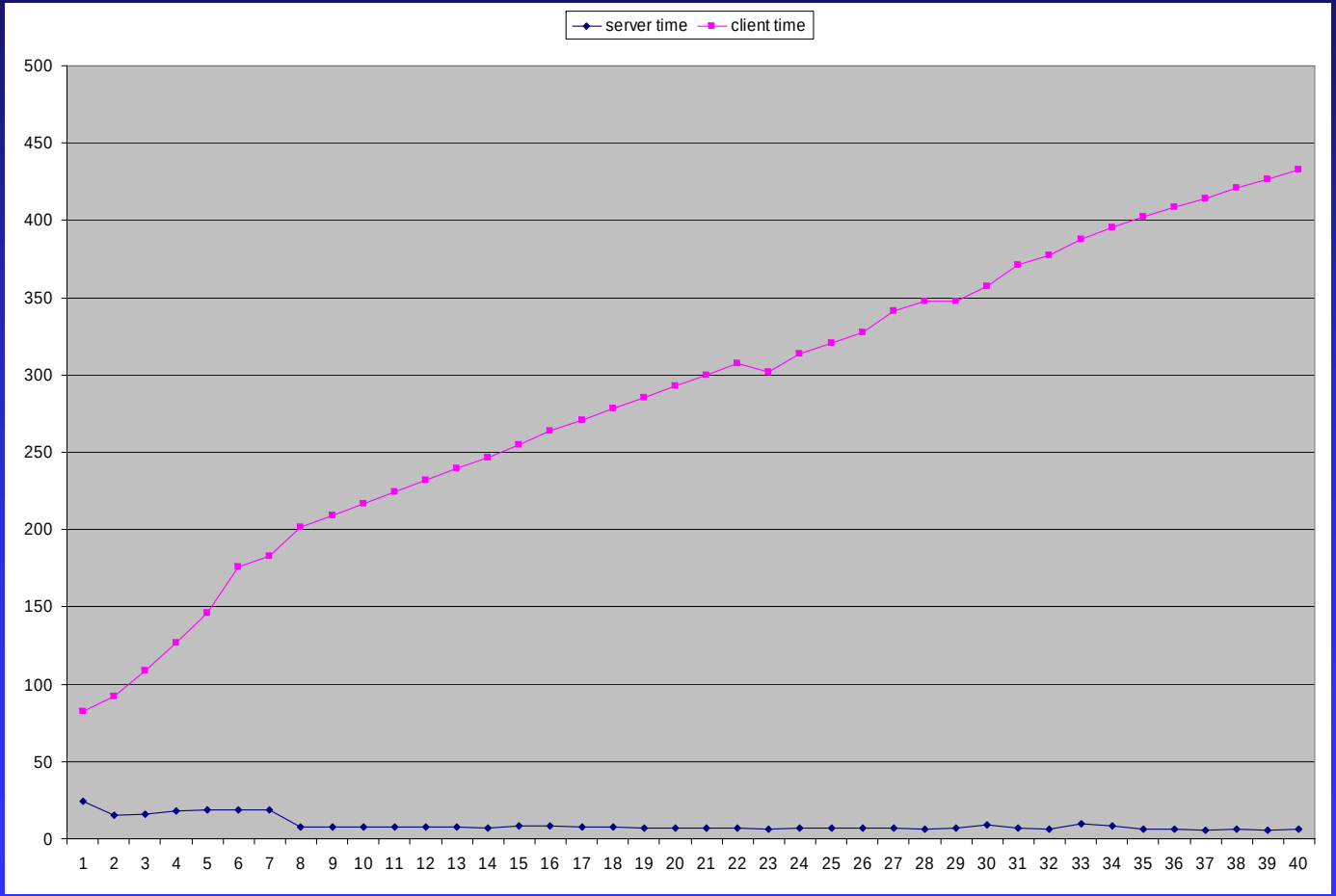


40 client hosts (FIP gateways), 0.5 Mb data, compression disabled

N	host	server time	client time
1	cfc-sr1-rl1h	8.17	89.01
2	cfc-sr1-rr1e	7.89	97.22
3	cfc-sr1-rr1i	6.88	105.21

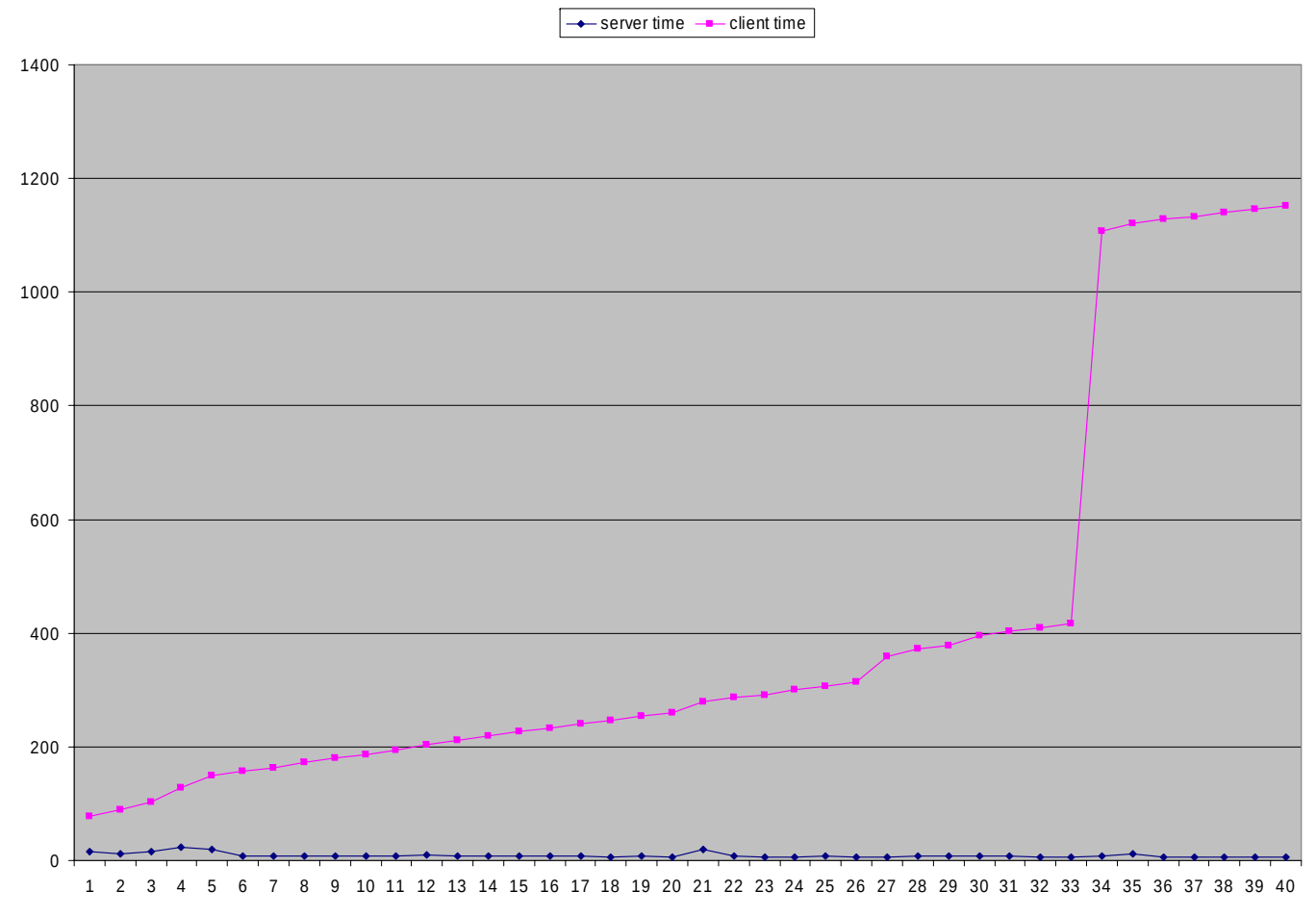
Scalability (2/4)

1 client/host, typical distribution



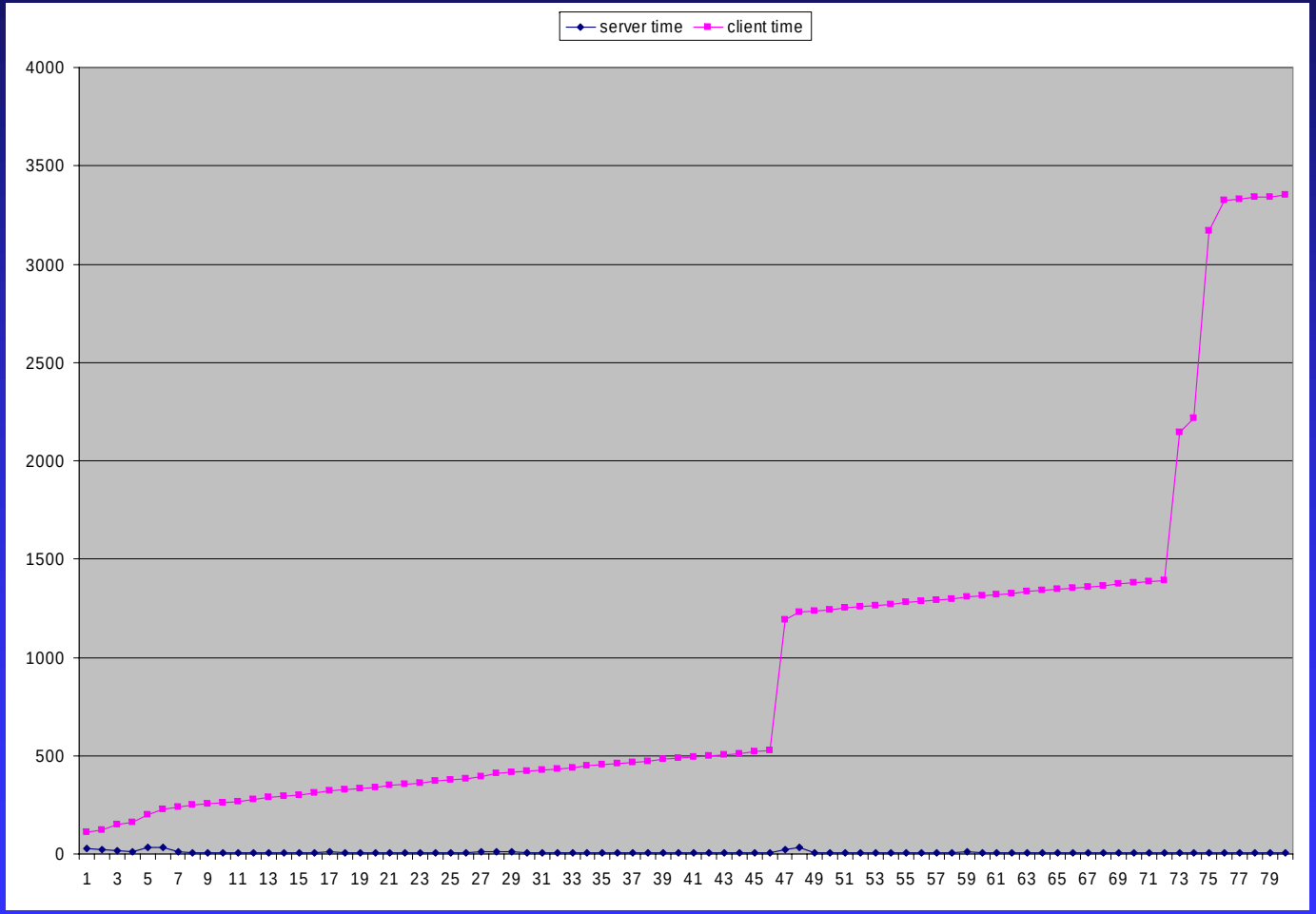
Scalability (3/4)

1 client/host, network congestion



Scalability (4/4)

2 clients/host, typical distribution



Current status

- Current results
  - PM server has been running non-stop during 2006 with typically ~60 clients connected. More than 100000 dumps, no data loss.
  - Operational in QPS and FGC. New clients to come soon: BI (BPM, BLM), RADMON, beam dump, RF.
  - No service interruption during the summer power cuts (~30 min), but
  - A short (~2 min) interruption during the CERN wide DNS failure in May (not noticed by users).
- TO DO
  - More scalability tests.
  - Multiple servers for fault tolerance and load balancing.