Western Norway
University of
Applied Sciences

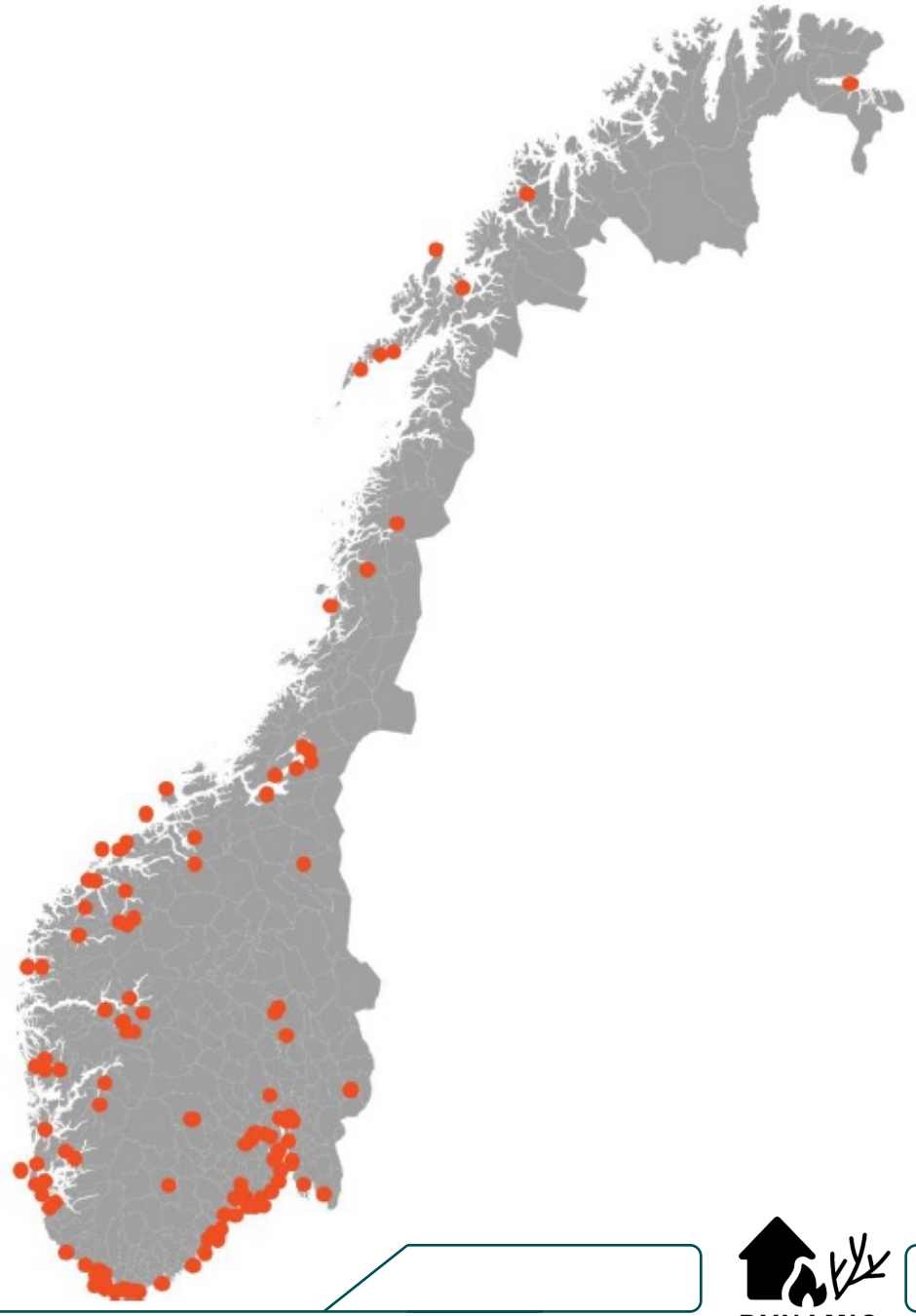
Formal Specification and Validation of a Data-driven Software System for Fire Risk Predictions

Ruben Dobler Strand[1], Laure Petrucci[2] and Lars Michael Kristensen[3]

[1] Department of Safety, Chemistry, and Biomedical laboratory sciences
Western Norway University of Applied Sciences, Haugesund, Norway.
[2] LIPN, CNRS UMR 7030, Université Sorbonne Paris Nord, Villetaneuse, France
[3] Department of Computer science, Electrical engineering and Mathematical sciences
Western Norway University of Applied Sciences, Bergen, Norway.

DYNAMIC

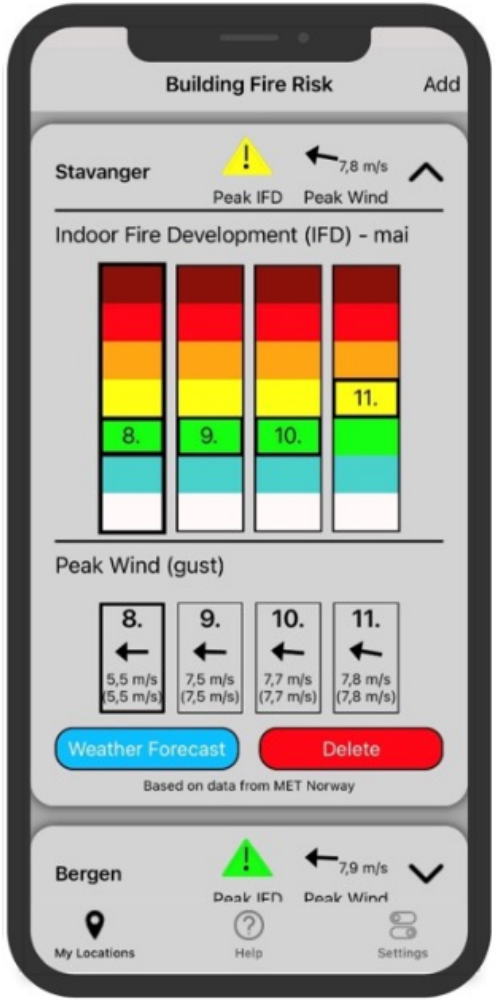
> 200 densely built wooden settlements worthy of preservation in Norway

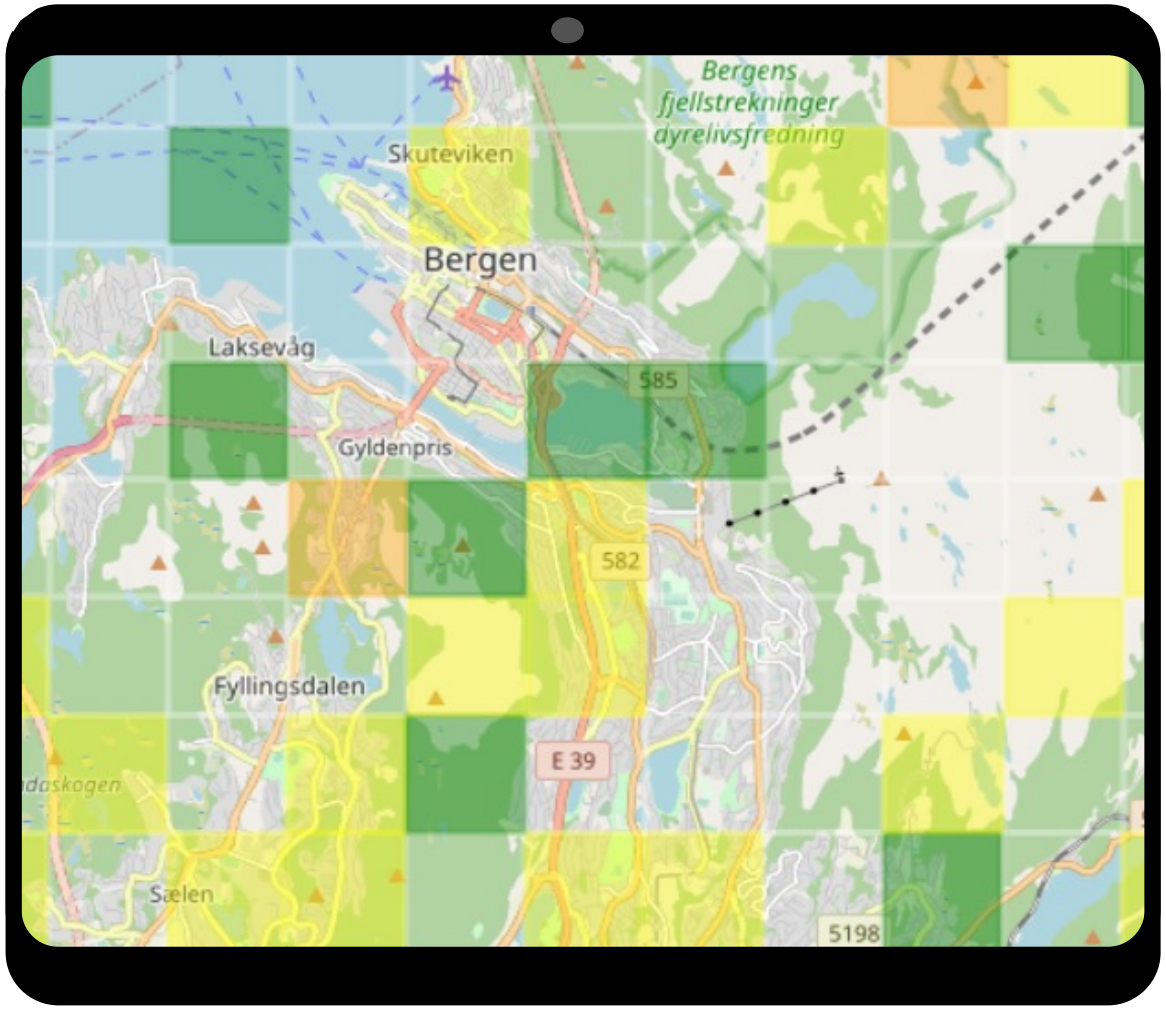


Predictive Fire Risk Notification System

CONCEPT

DEVELOPED GUI





Predictive fire risk indication

[Log (2017), Log (2018), Log(2019)]

Based on measurements of the outdoor climate

- Temperature (measured)
- Relative humidity (measured)

Estimating indoor climate

- Air change rate / ventilation (computed) $\dot{m}_{AC}$
- Moisture release from enclosure (computed) $\dot{m}_{wall}$
- Internal moisture production (from literature) $\dot{m}_{supply}$
- Indoor water concentration in air (computed) $\mathrm{V}\frac{dC}{dt}$
- Relative humidity (computed)

Estimate water concentration in each of the layers of wood

- Compute Fuel Moisture Content (FMC)
- Compute fire risk indication as the time to flashover (TTF)

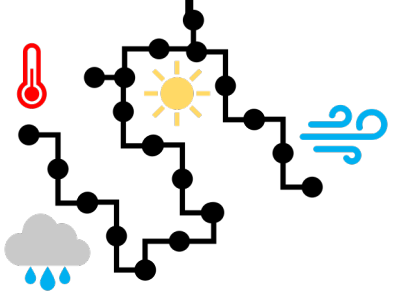
Røros Credit: Gjermund Søreng

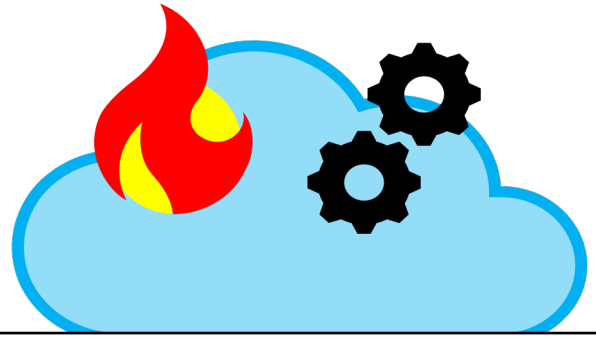
Context and overview of system

Weather data measurements and forecasting cloud-services

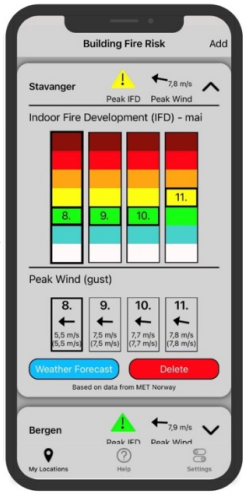
Meteorologisk institutt

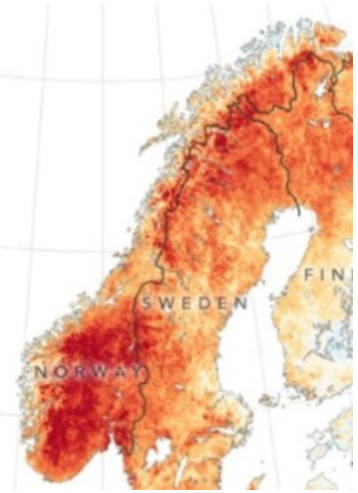
Frost





Distributed cloud-based fire risk notification system





Consumer-grade IoT weather station services

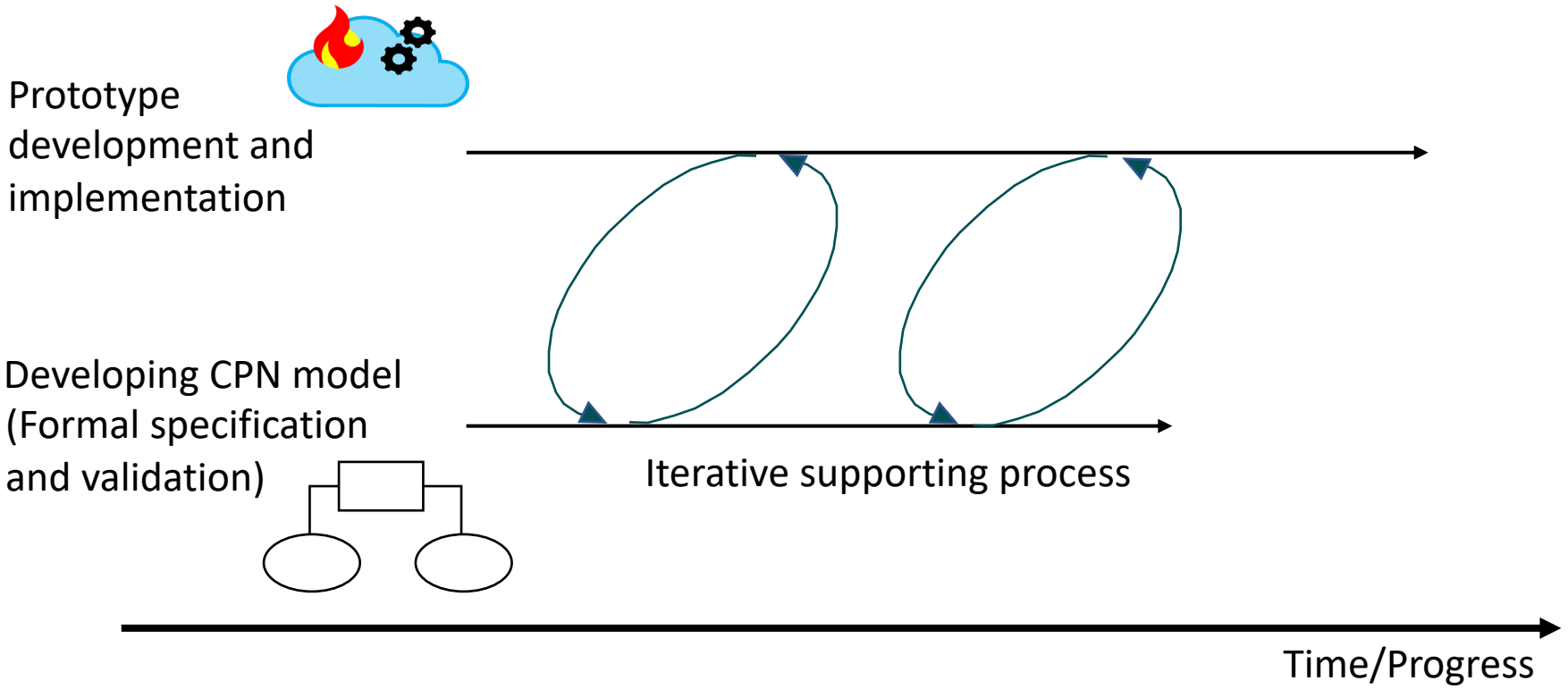
Fire risk models
[Log(2019)]

$$V\frac{\mathrm{d}C}{\mathrm{d}t} + \dot{m}_{\mathrm{AC}} + \dot{m}_{\mathrm{wall}} + \dot{m}_{\mathrm{supply}} = \mathbf{0}$$

DYNAMIC

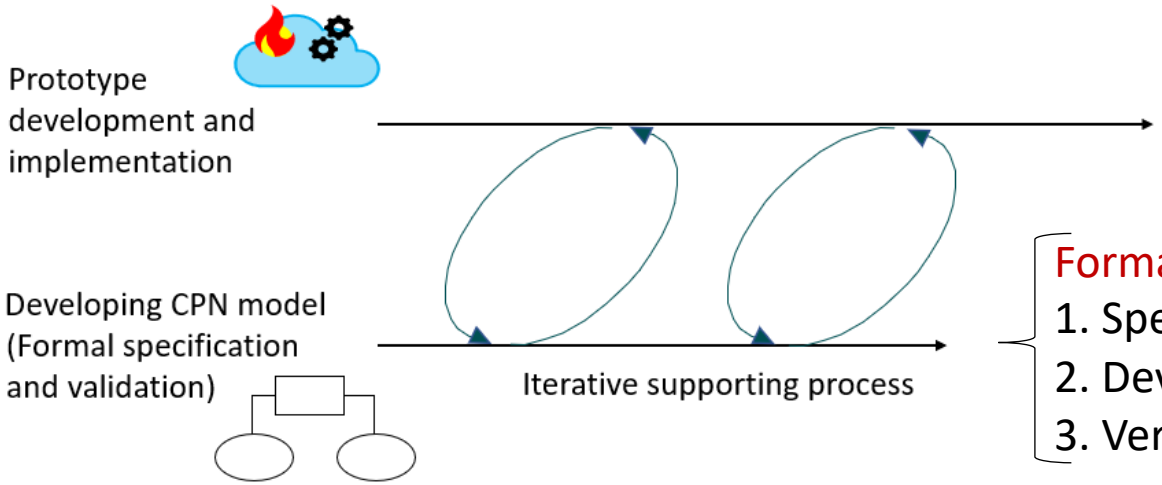
Development Process & Research Aims

Prototype development/Implementation in parallell with CPN model development



Development Process & Research Aims

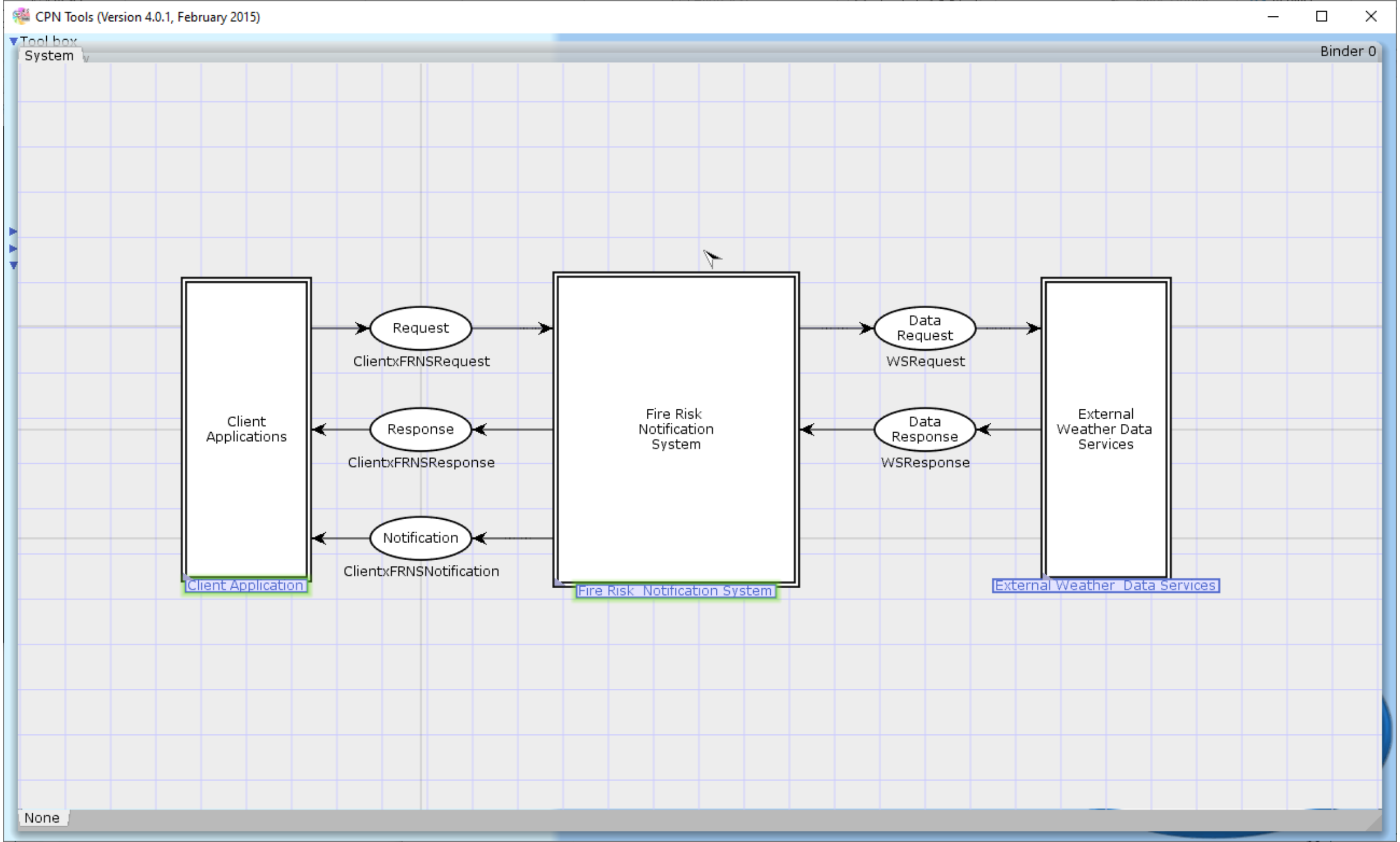
Prototype development/Implementation in parallell with CPN model development

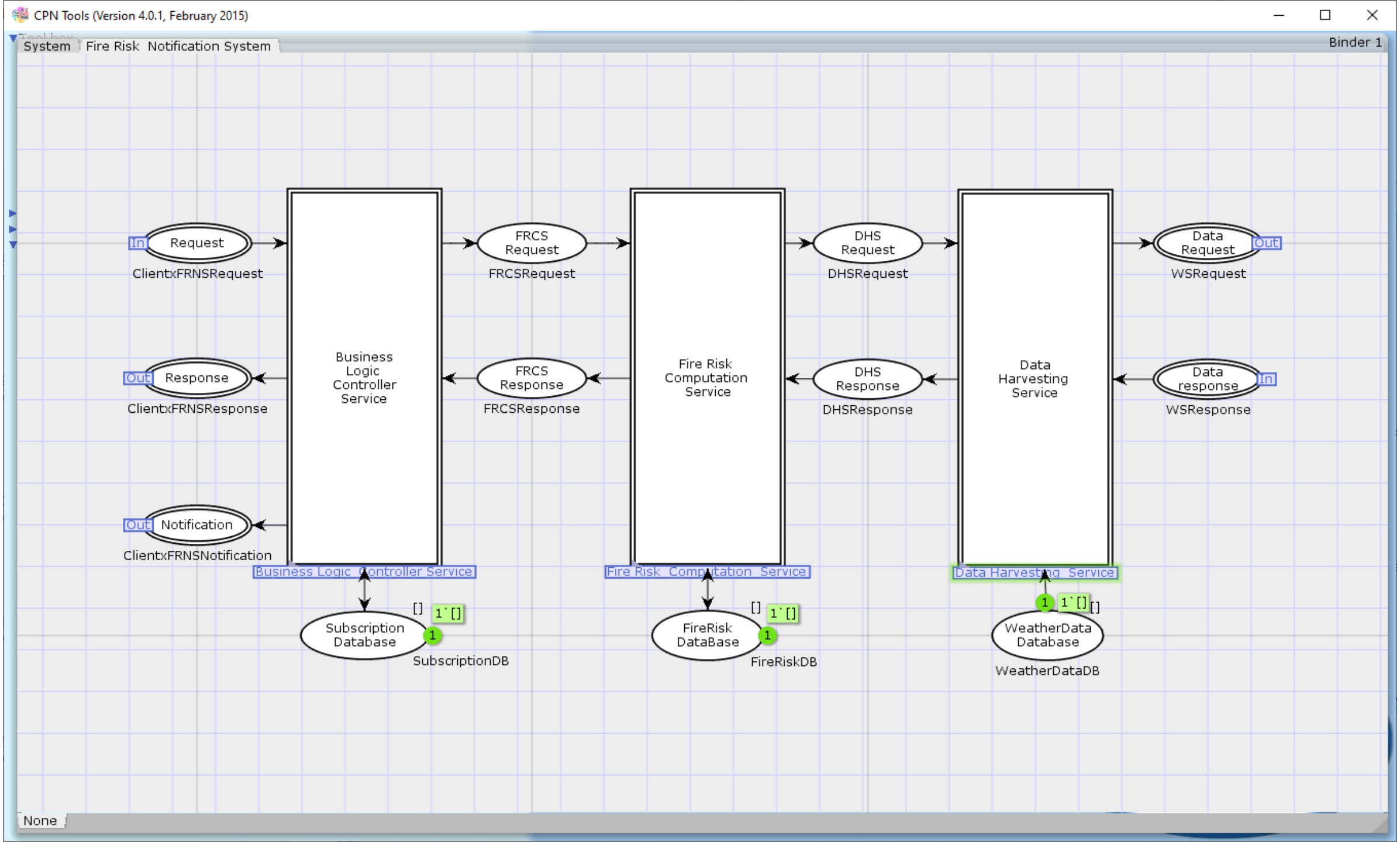


Validating the predictive fire risk indication model
Prototyping a fire risk indication service

Formal modelling of the software system
1. Specify software architecture and constituent micro-services
2. Develop modelling patterns for REST-based micro-services
3. Verify system services and behavioural interaction of micro-services

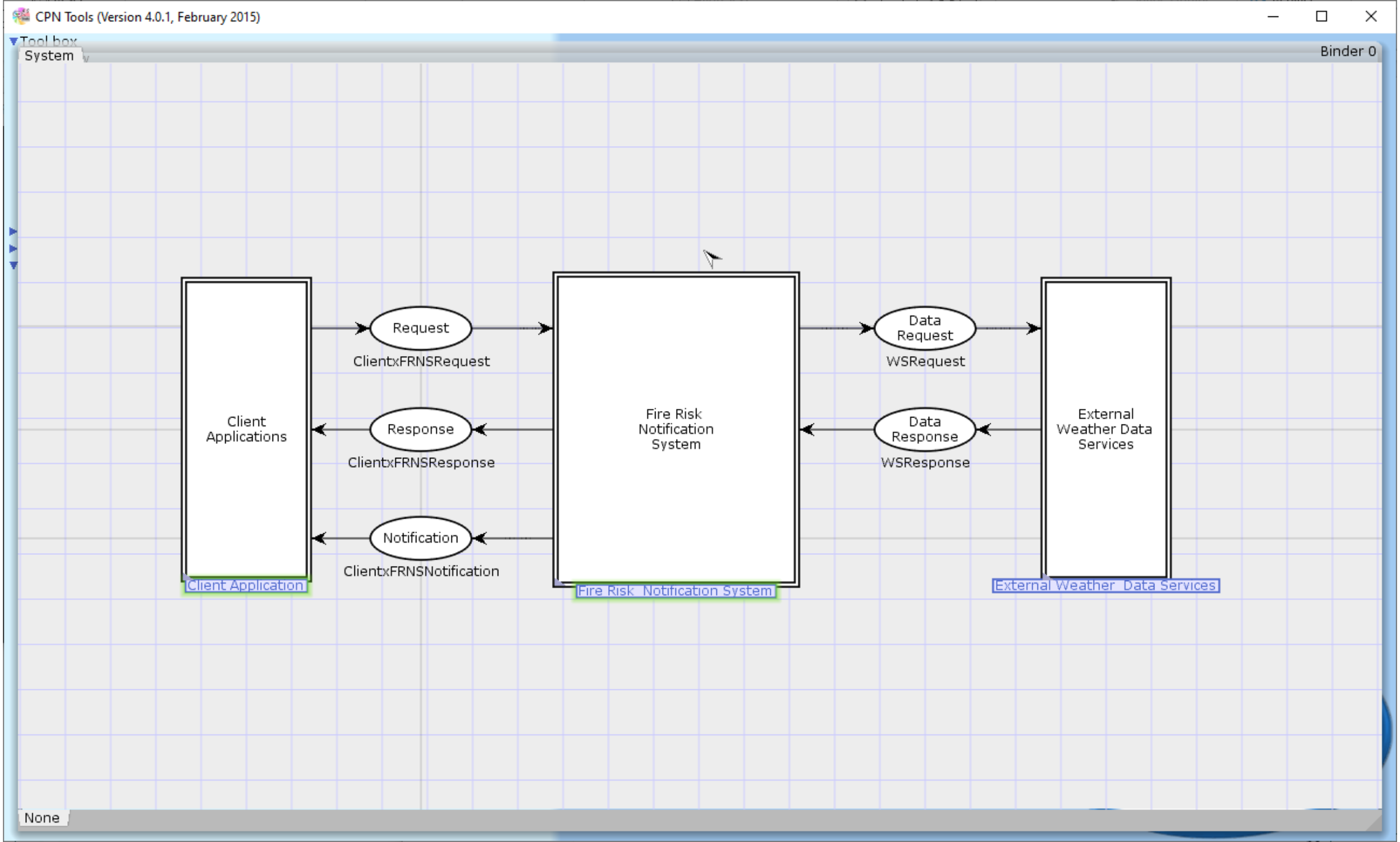
CPN model presentation

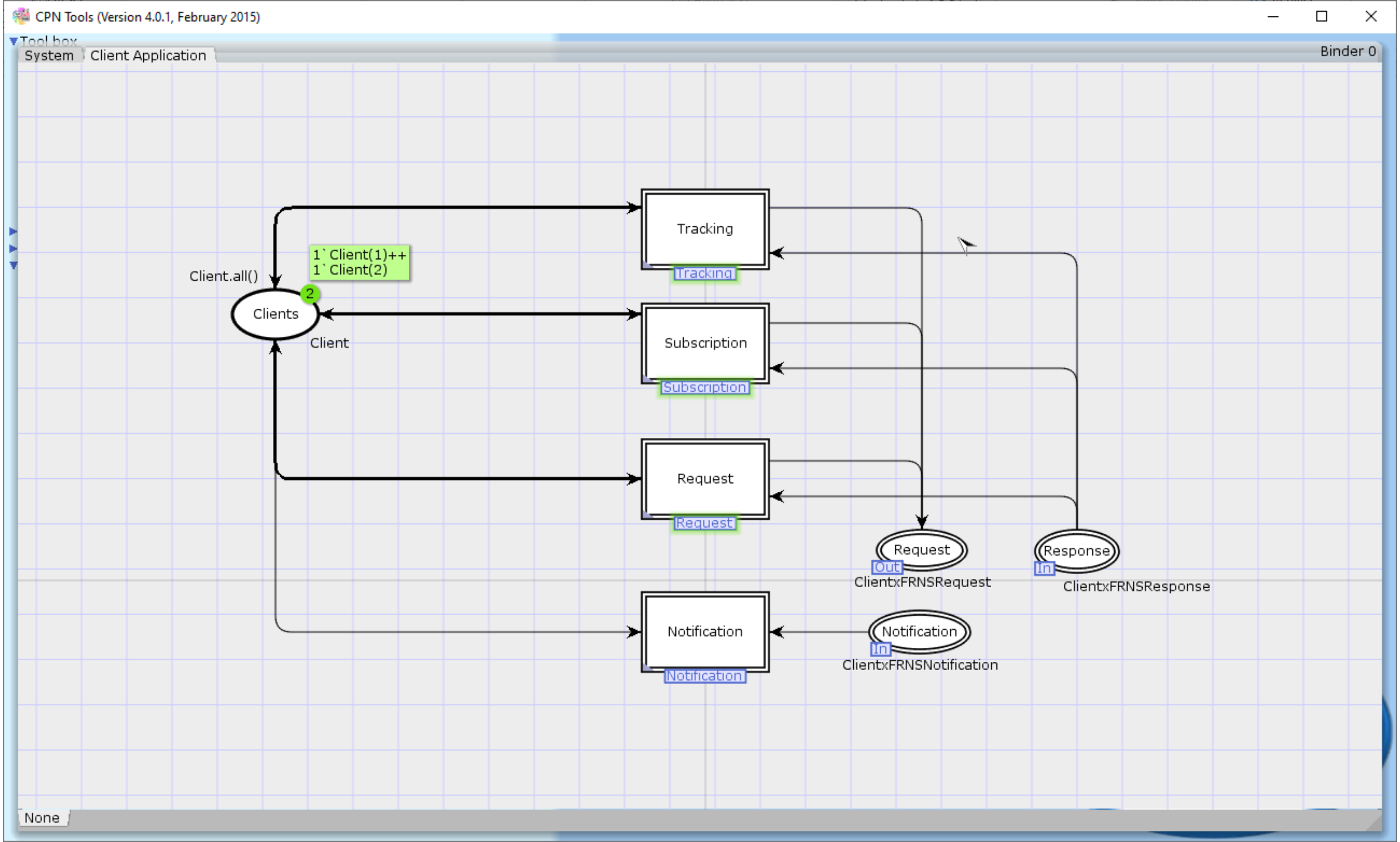

CPN Tools (Version 4.0.1, February 2015)
Tool box
System
Binder 0
Client Applications
Request
ClientxFRNSRequest
Response
ClientxFRNSResponse
Notification
ClientxFRNSNotification
Client Application
Fire Risk Notification System
Fire Risk_Notification System
Data Request
WSRequest
Data Response
WSResponse
External Weather Data Services
External Weather_Data Services
None

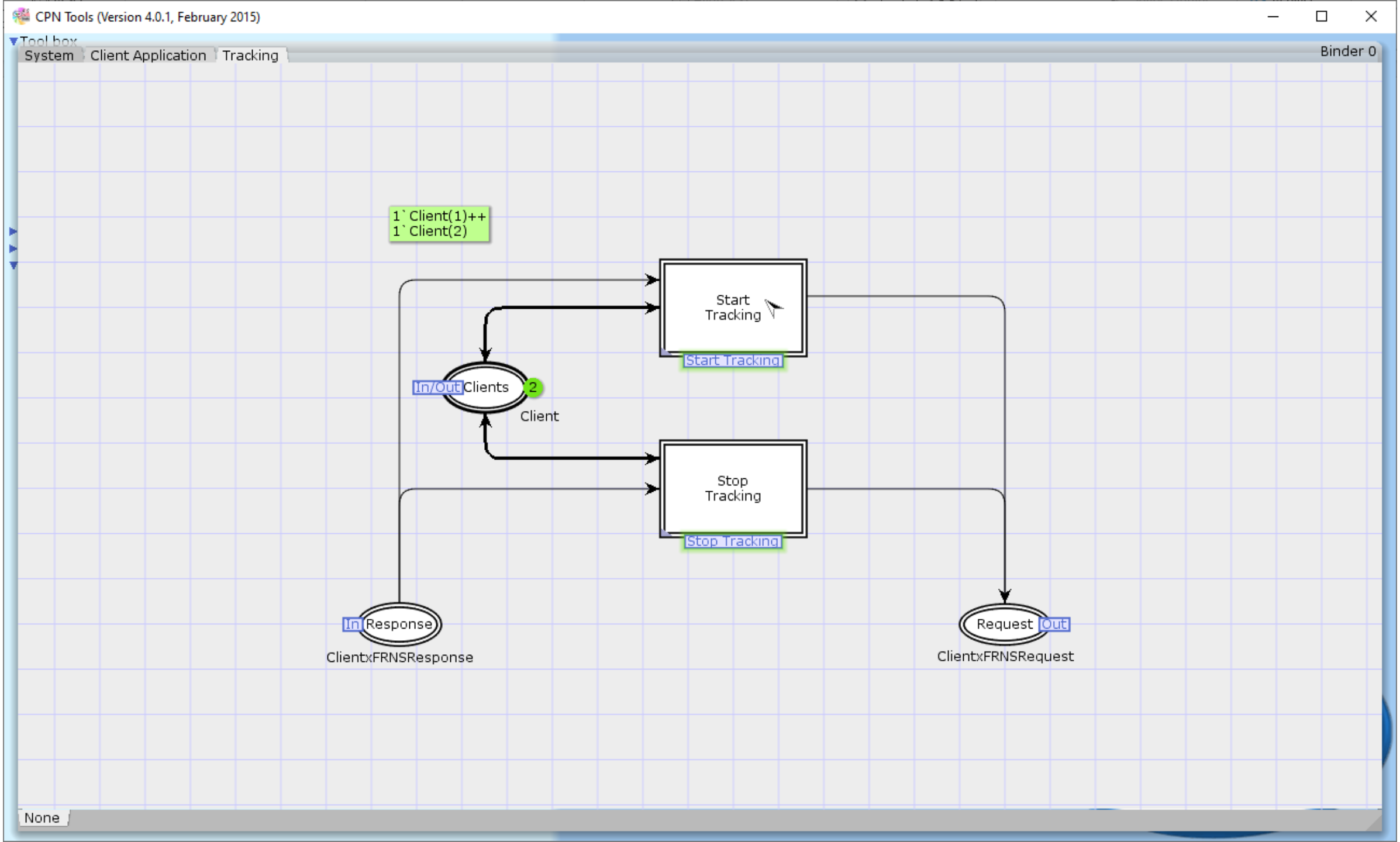

CPN Tools (Version 4.0.1, February 2015)
System
Fire Risk Notification System
Binder 1
In
Request
ClientxFRNSRequest
Business Logic Controller Service
FRCS Request
FRCSRequest
Fire Risk Computation Service
DHS Request
DHSRequest
Data Harvesting Service
Data Request
Out
WSRequest
Out
Response
ClientxFRNSResponse
FRCS Response
FRCSResponse
DHS Response
DHSResponse
Data response
In
WSResponse
Out
Notification
ClientxFRNSNotification
Business Logic Controller Service
Fire Risk Computation Service
Data Harvesting Service
Subscription Database
[]
1`[]
1
SubscriptionDB
FireRisk DataBase
[]
1`[]
1
FireRiskDB
1
1`[]
[]
WeatherData Database
WeatherDataDB
None

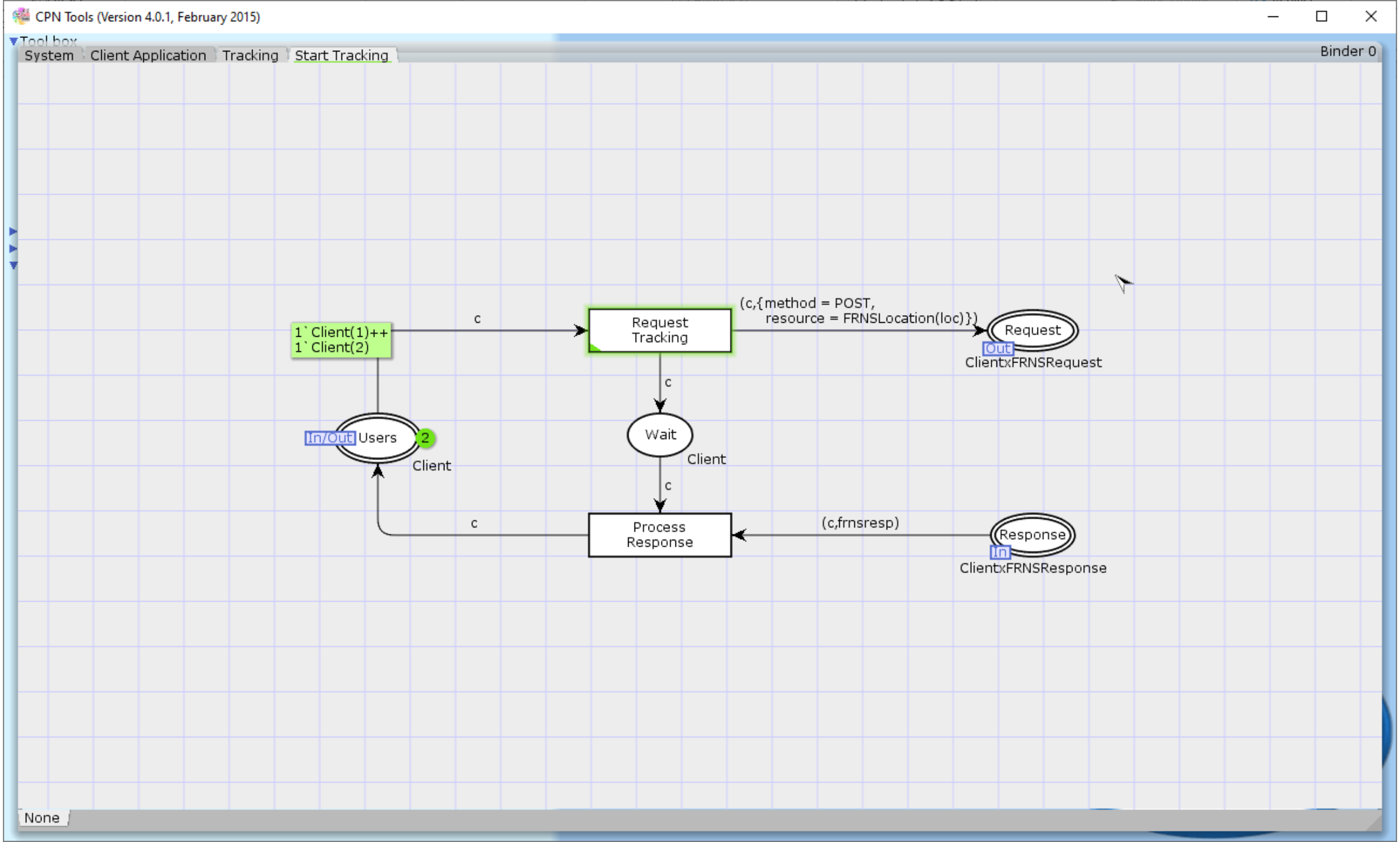
CPN model presentation

Example


CPN Tools (Version 4.0.1, February 2015)
Tool box
System
Binder 0
Client Applications
Request
ClientxFRNSRequest
Response
ClientxFRNSResponse
Notification
ClientxFRNSNotification
Client Application
Fire Risk Notification System
Fire Risk_Notification System
Data Request
WSRequest
Data Response
WSResponse
External Weather Data Services
External Weather_Data Services
None

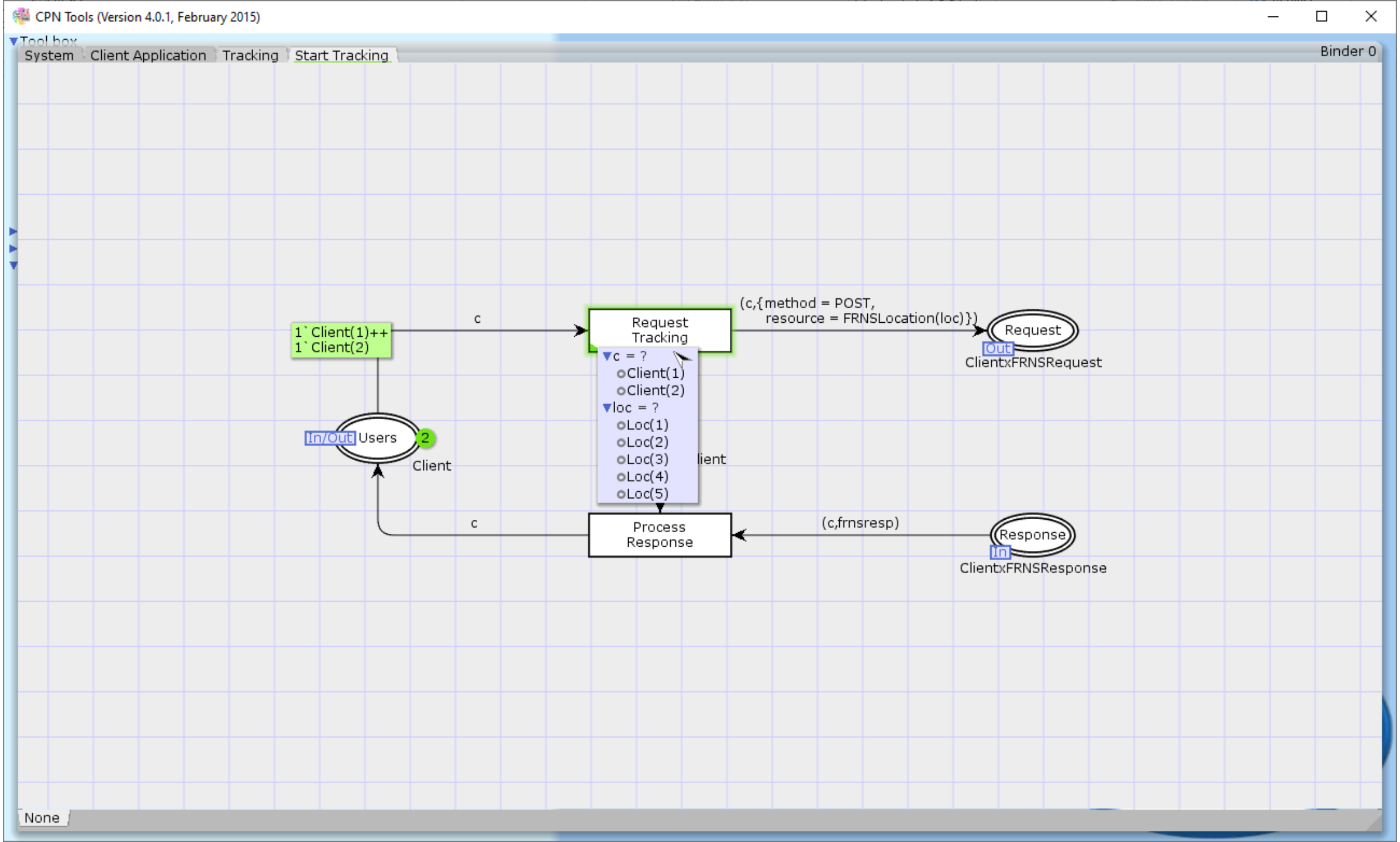

CPN Tools (Version 4.0.1, February 2015)
Tool box
System
Client Application
Binder 0
Client.all()
1`Client(1)++
1`Client(2)
2
Clients
Client
Tracking
Tracking
Subscription
Subscription
Request
Request
Notification
Notification
Request
Out
ClientxFRNSRequest
Response
In
ClientxFRNSResponse
Notification
In
ClientxFRNSNotification
None

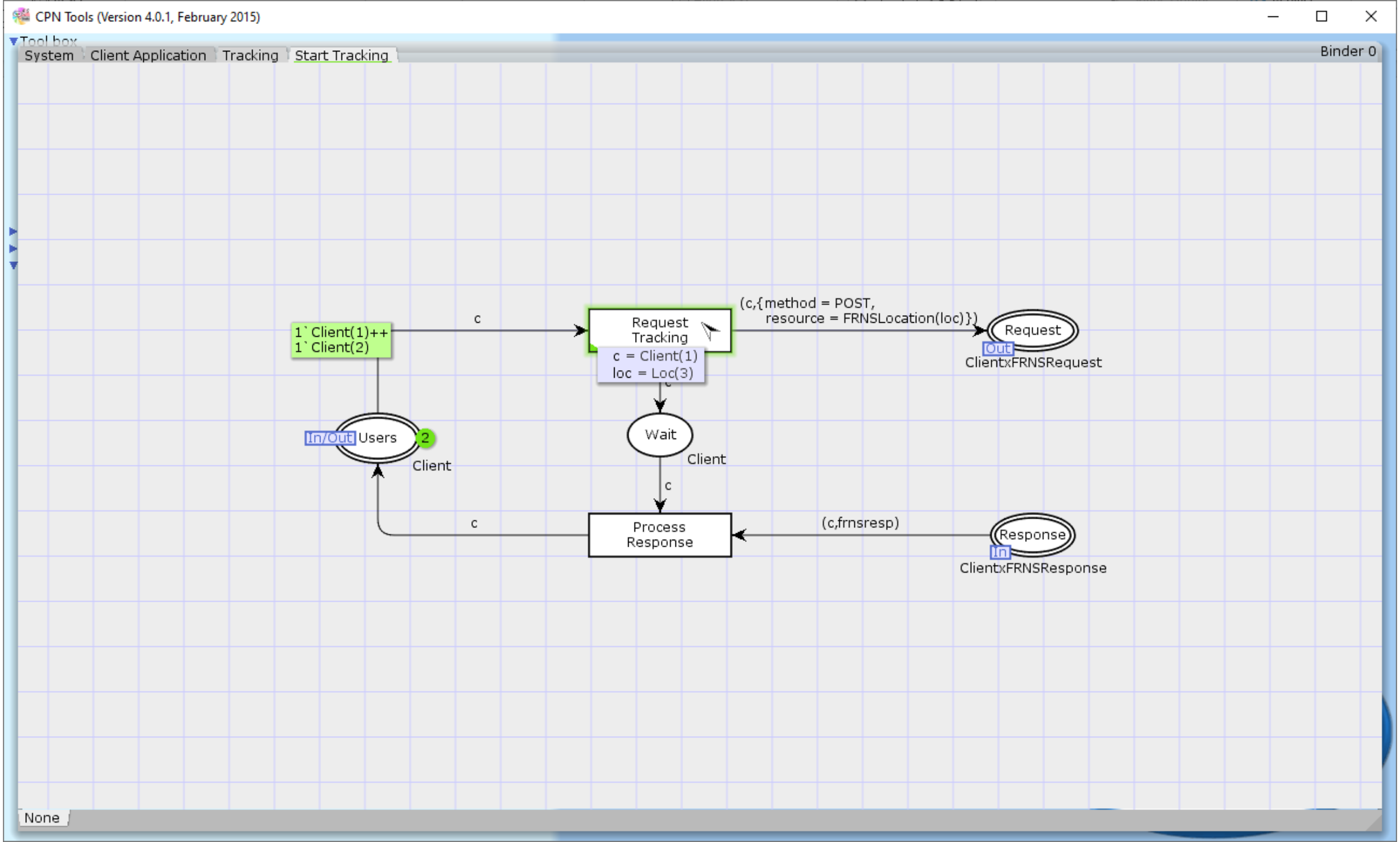

CPN Tools (Version 4.0.1, February 2015)
Tool box
System
Client Application
Tracking
Binder 0
1`Client(1)++
1`Client(2)
Start
Tracking
Start Tracking
In/Out
Clients
2
Client
Stop
Tracking
Stop Tracking
In
Response
ClientxFRNSResponse
Request
Out
ClientxFRNSRequest
None

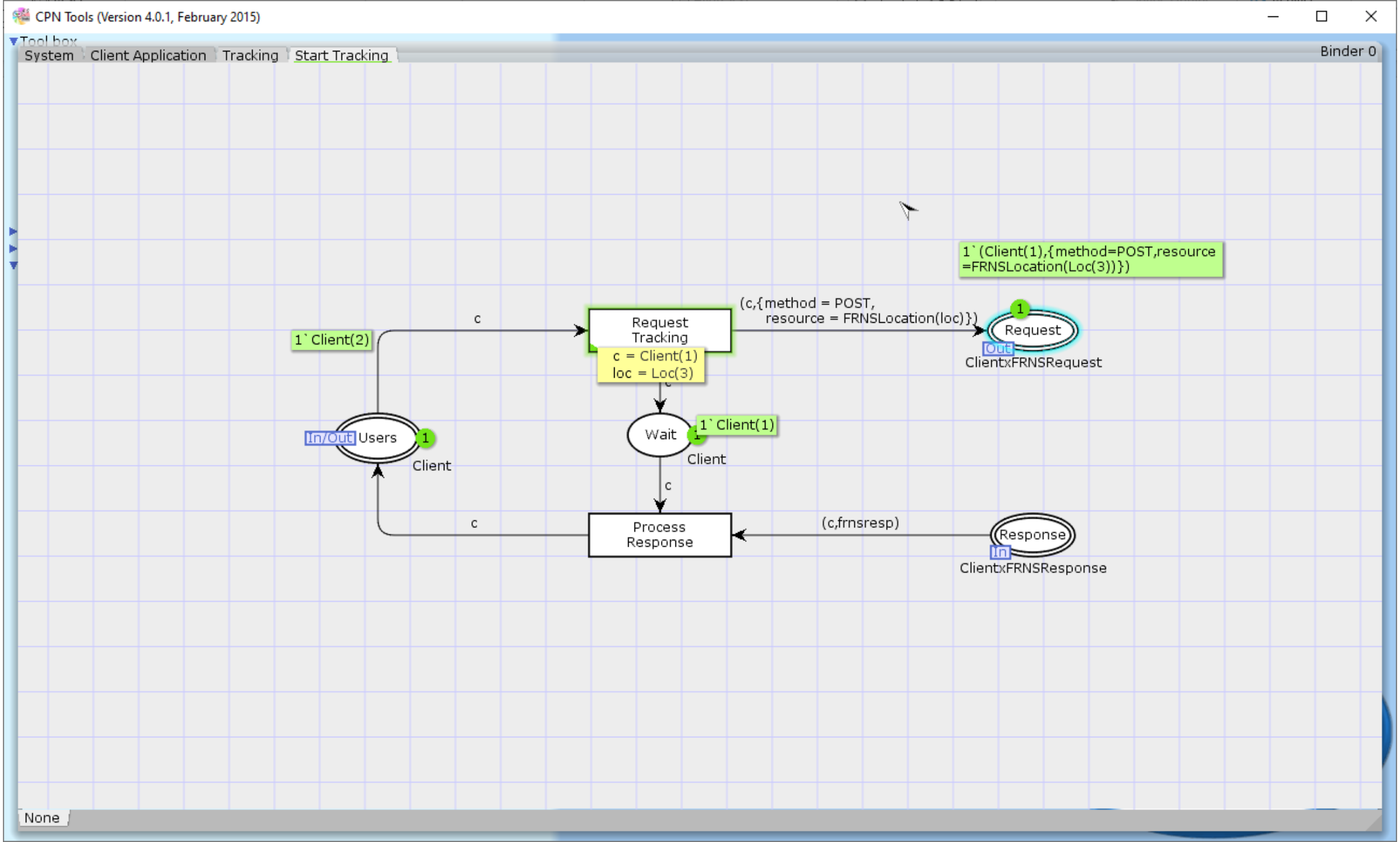

CPN Tools (Version 4.0.1, February 2015)
Tool box
System
Client Application
Tracking
Start Tracking
Binder 0
1`Client(1)++
1`Client(2)
c
Request
Tracking
(c,{method = POST,
resource = FRNSLocation(loc)})
Request
Out
ClientxFRNSRequest
c
In/Out
Users
2
Client
Wait
Client
c
c
Process
Response
(c,frnsresp)
Response
In
ClientxFRNSResponse
None

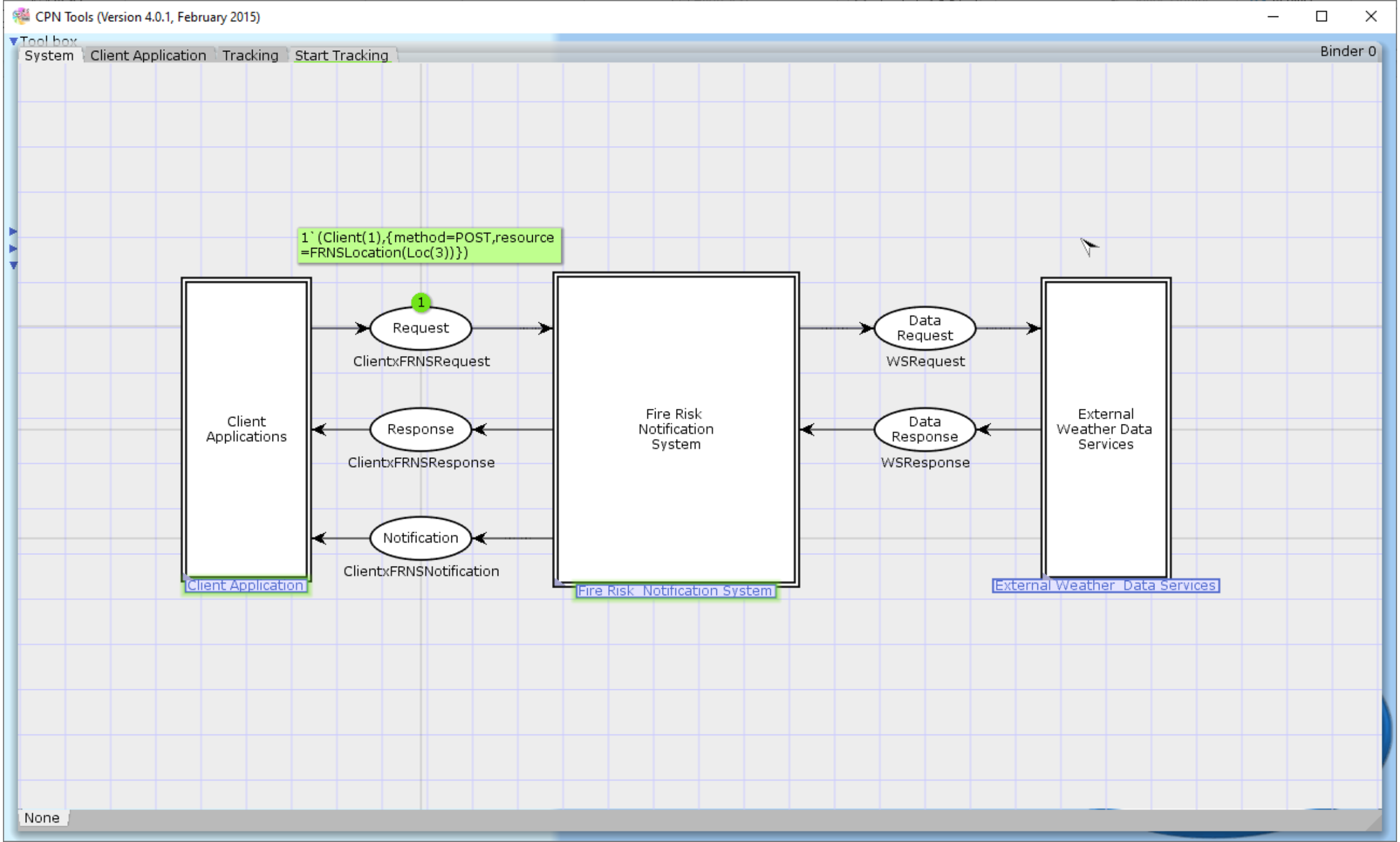
```
1  colset Method        = with  GET | PUT      | POST         | DELETE;
2  colset StatusCode    = with  OK  | CREATED  | ACCEPTED     | NOTFOUND;
3  colset Component     = union BLCSTracking  + BLCSUpdate + BLCSRequest;
4
5  val Cn = 2;
6  val Ln = 5;
7  colset Client          = index   Client with 1..Cn;
8  colset Location        = index   Loc    with 1..Ln;
9  colset Locations       = list    Location;
10 colset ClientxLocation = product Client * Location;
11
12 colset FireRisk      = with    Risk | NA;
13 colset LocxFireRisk = product Location * FireRisk;
14 colset FireRisks     = list    LocxFireRisk;
15 colset FRCSResource = union   FRCSFirerisks + FRCSLocation : Location;
16 colset FRNSResource = union   FRNSLocation  : Location        + FRNSFirerisk  : Location +
17                               FRNSLocations : ClientxLocation + FRNSFirerisks : FireRisks;
18
19 colset FRNSRequest          = record method : Method * resource : FRNSResource;
20 colset ClientxFRNSRequest   = product Client * FRNSRequest;
21
22 colset FRNSResponse         = record  response : StatusCode * body : FRNSResource;
23 colset ClientxFRNSResponse = product Client   * FRNSResponse;
```

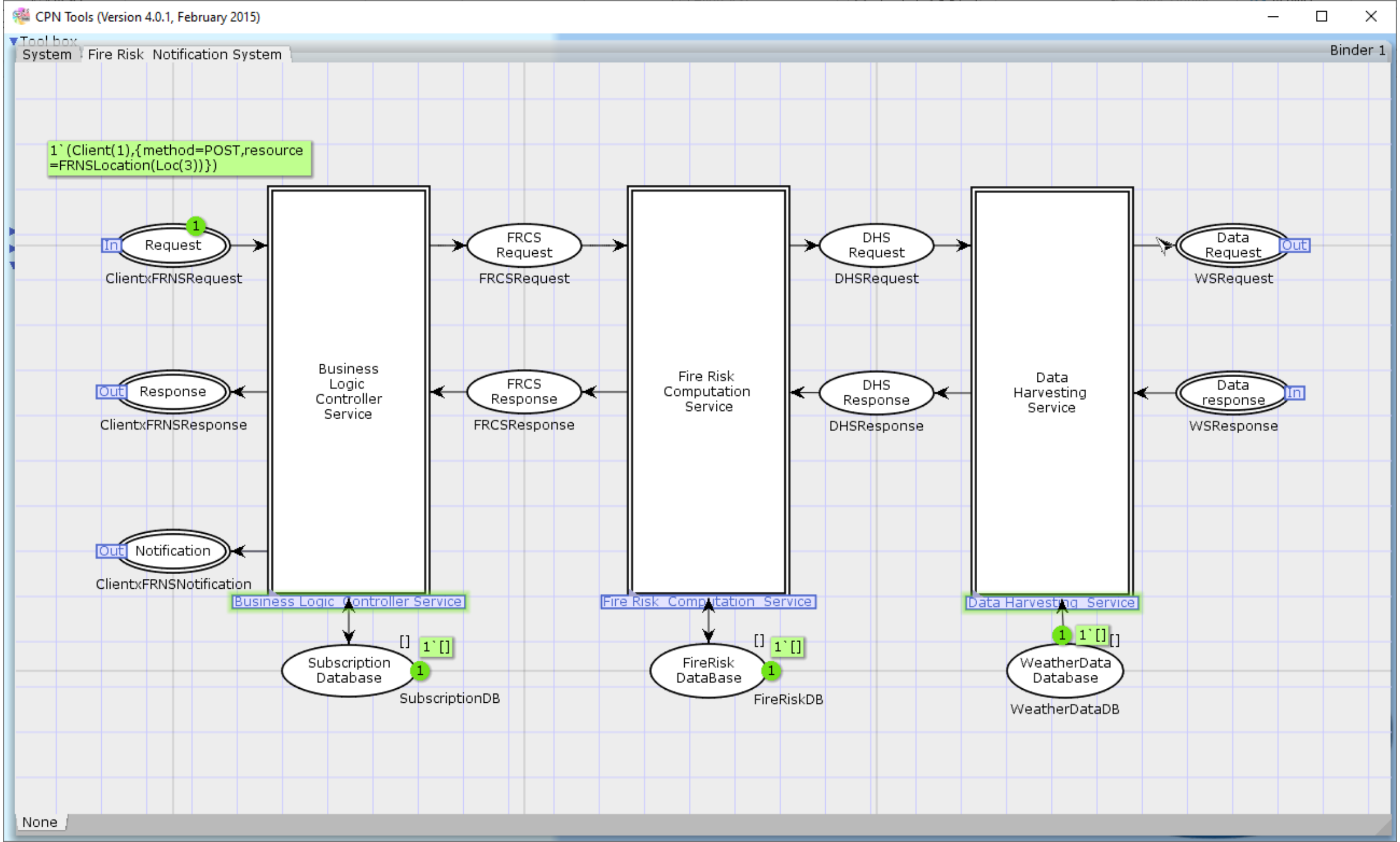
```
1  colset Method       = with  GET | PUT      | POST         | DELETE;
2  colset StatusCode   = with  OK  | CREATED  | ACCEPTED     | NOTFOUND;
3  colset Component    = union BLCSTracking  + BLCSUpdate + BLCSRequest;
4
5  val Cn = 2;
6  val Ln = 5;
7  colset Client          = index   Client with 1..Cn;
8  colset Location        = index   Loc    with 1..Ln;
9  colset Locations       = list    Location;
10 colset ClientxLocation = product Client * Location;
11
12 colset FireRisk     = with    Risk | NA;
13 colset LocxFireRisk = product Location * FireRisk;
14 colset FireRisks    = list    LocxFireRisk;
15 colset FRCSResource = union   FRCSFirerisks + FRCSLocation : Location;
16 colset FRNSResource = union   FRNSLocation  : Location        + FRNSFirerisk  : Location +
17                               FRNSLocations : ClientxLocation + FRNSFirerisks : FireRisks;
18
19 colset FRNSRequest          = record  method : Method * resource : FRNSResource;
20 colset ClientxFRNSRequest   = product Client * FRNSRequest;
21
22 colset FRNSResponse         = record  response : StatusCode * body : FRNSResource;
23 colset ClientxFRNSResponse = product Client   * FRNSResponse;
```

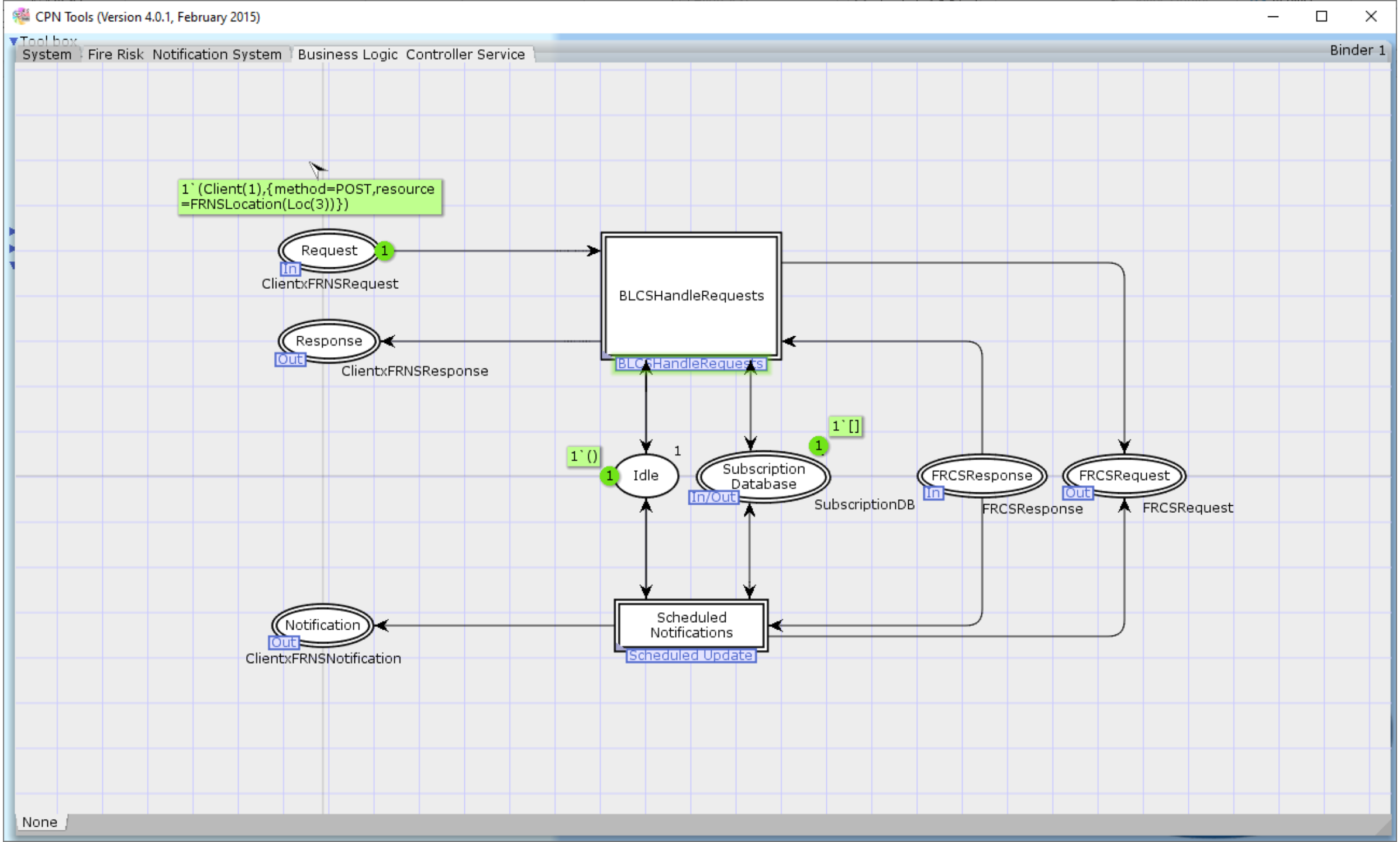

CPN Tools (Version 4.0.1, February 2015)
Tool box
System
Client Application
Tracking
Start Tracking
Binder 0
1`Client(1)++
1`Client(2)
c
Request
Tracking
(c,{method = POST,
resource = FRNSLocation(loc)})
Request
Out
ClientxFRNSRequest
c = ?
Client(1)
Client(2)
loc = ?
Loc(1)
Loc(2)
Loc(3)
Loc(4)
Loc(5)
In/Out
Users
2
Client
c
Process
Response
(c,frnsresp)
Response
In
ClientxFRNSResponse
None

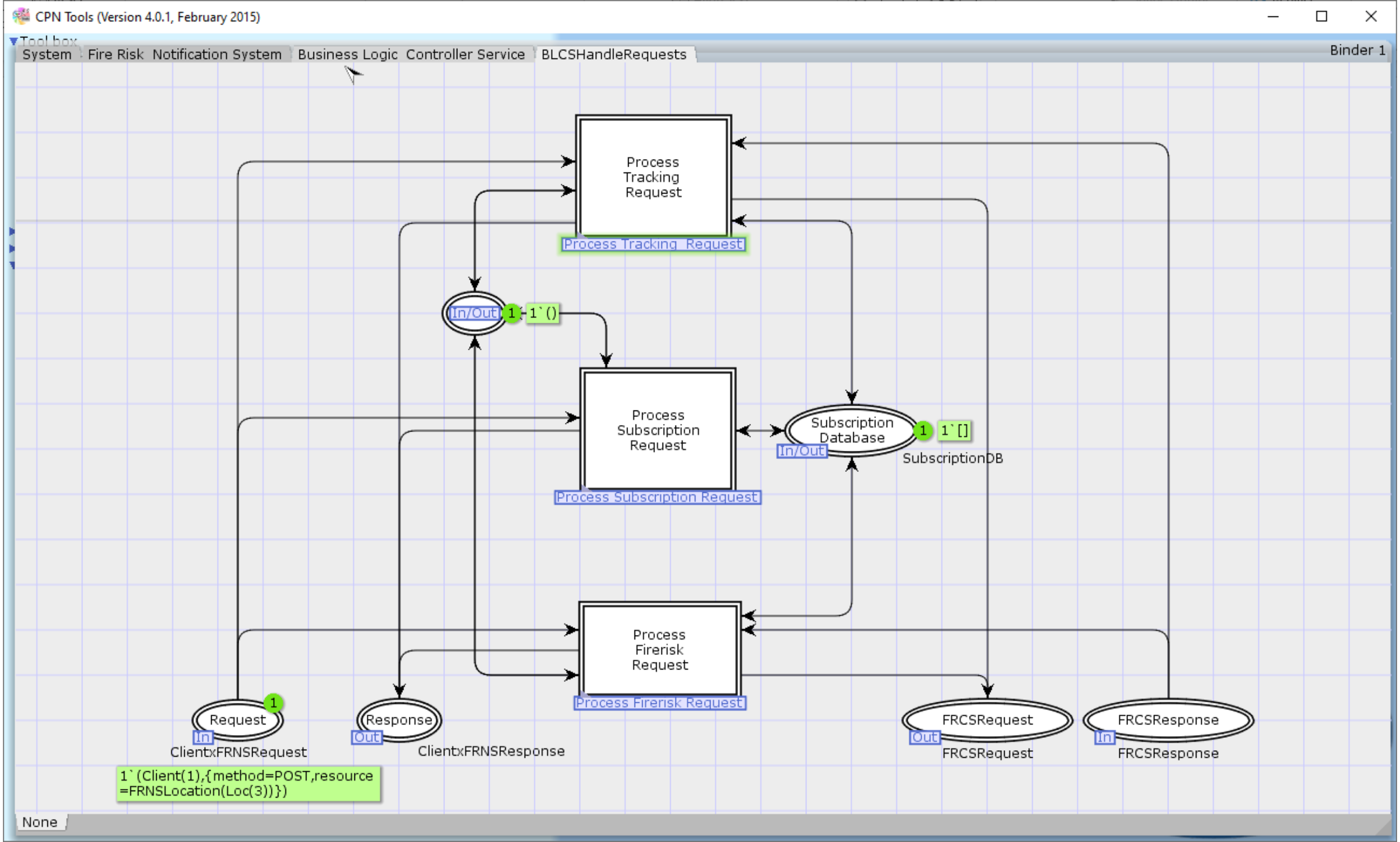

CPN Tools (Version 4.0.1, February 2015)
Tool box
System
Client Application
Tracking
Start Tracking
Binder 0
1`Client(1)++
1`Client(2)
c
Request
Tracking
c = Client(1)
loc = Loc(3)
(c,{method = POST,
resource = FRNSLocation(loc)})
Request
Out
ClientxFRNSRequest
In/Out
Users
2
Client
Wait
Client
c
c
Process
Response
(c,frnsresp)
Response
In
ClientxFRNSResponse
None

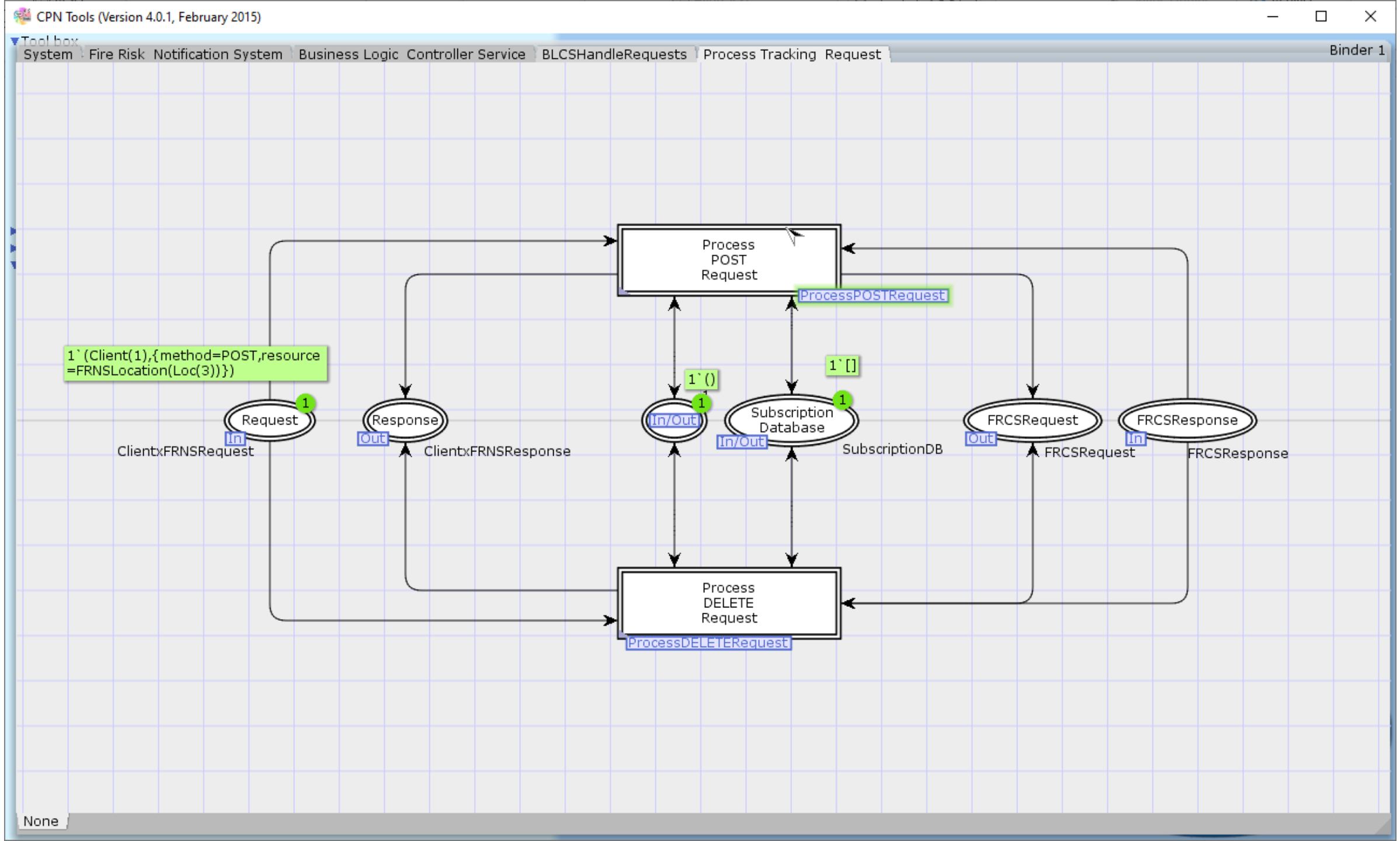

CPN Tools (Version 4.0.1, February 2015)
Tool box
System
Client Application
Tracking
Start Tracking
Binder 0
1`(Client(1),{method=POST,resource =FRNSLocation(Loc(3))})
(c,{method = POST, resource = FRNSLocation(loc)})
c
1`Client(2)
Request Tracking
c = Client(1)
loc = Loc(3)
1
Request
Out
ClientxFRNSRequest
In/Out
Users
1
Client
Wait
1`Client(1)
Client
c
c
Process Response
(c,frnsresp)
Response
In
ClientxFRNSResponse
None

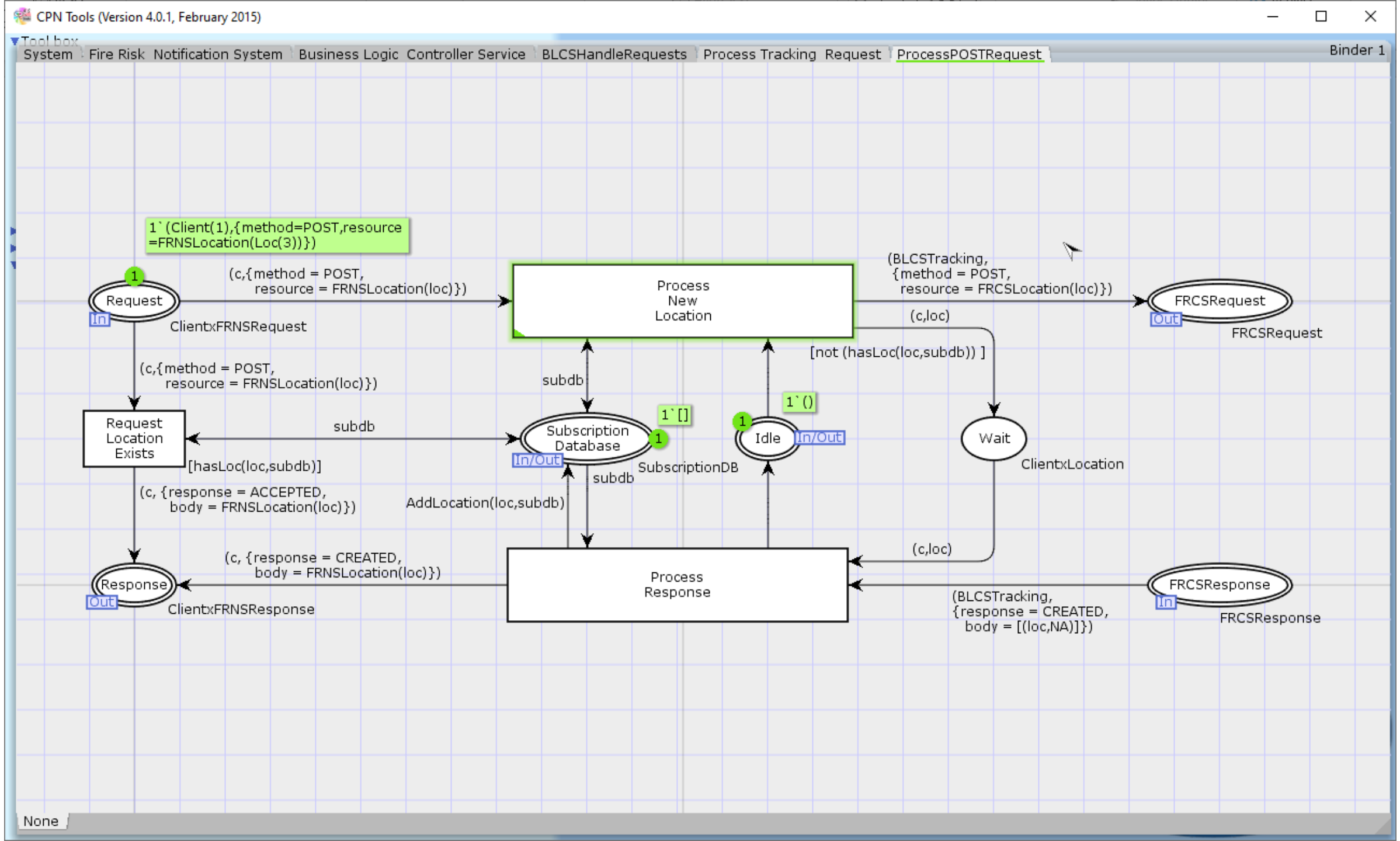

CPN Tools (Version 4.0.1, February 2015)
Tool box
System
Client Application
Tracking
Start Tracking
Binder 0
1`(Client(1),{method=POST,resource =FRNSLocation(Loc(3))})
1
Request
ClientxFRNSRequest
Response
ClientxFRNSResponse
Notification
ClientxFRNSNotification
Client Applications
Client Application
Fire Risk Notification System
Fire Risk_Notification System
Data Request
WSRequest
Data Response
WSResponse
External Weather Data Services
External Weather_Data Services
None

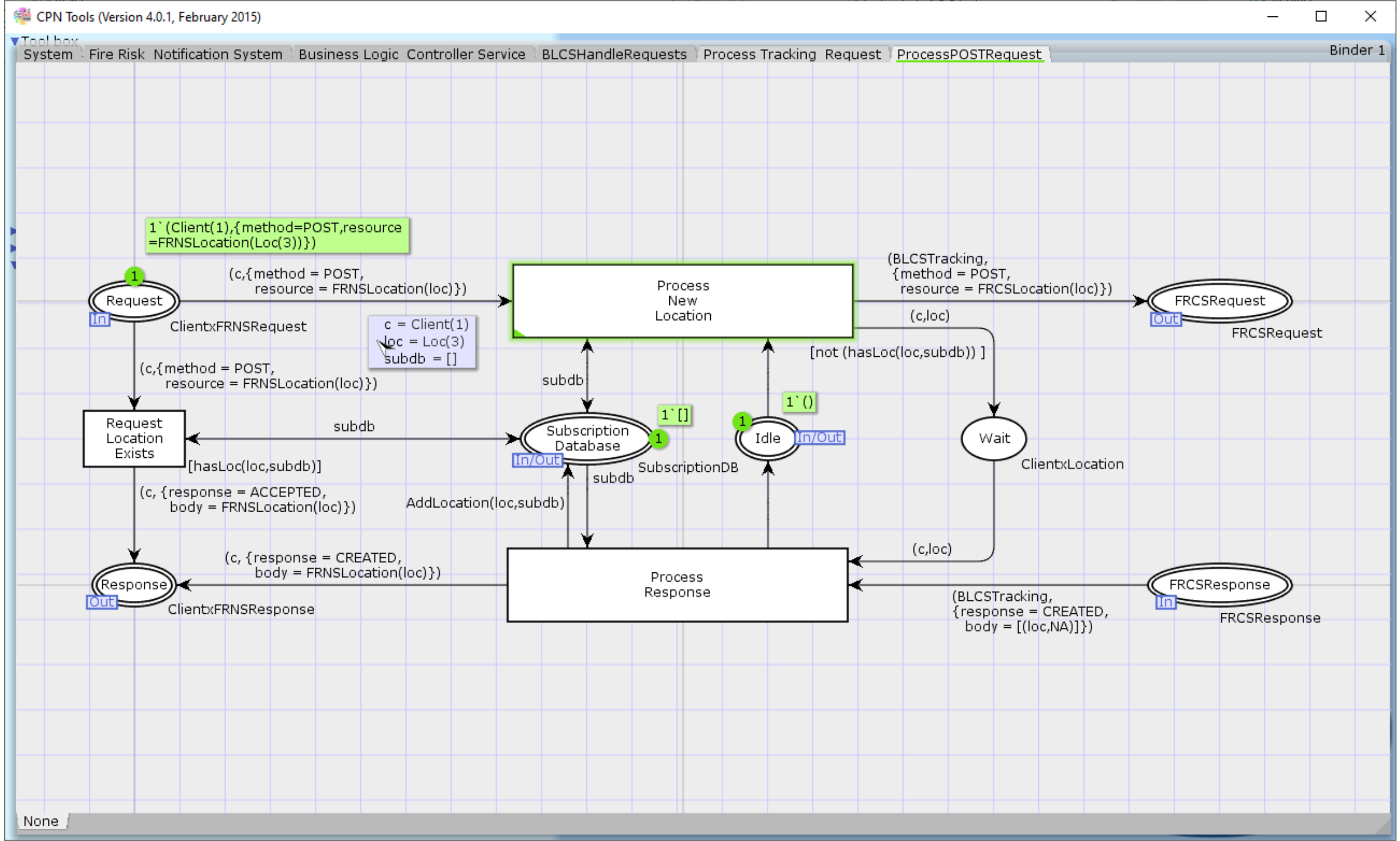

CPN Tools (Version 4.0.1, February 2015)
Tool box
System
Fire Risk Notification System
Binder 1
1`(Client(1),{method=POST,resource =FRNSLocation(Loc(3))})
In
Request
ClientxFRNSRequest
FRCS Request
FRCSRequest
DHS Request
DHSRequest
Data Request
Out
WSRequest
Out
Response
ClientxFRNSResponse
Business Logic Controller Service
FRCS Response
FRCSResponse
Fire Risk Computation Service
DHS Response
DHSResponse
Data Harvesting Service
Data response
In
WSResponse
Out
Notification
ClientxFRNSNotification
Business Logic Controller Service
Fire Risk Computation Service
Data Harvesting Service
Subscription Database
SubscriptionDB
FireRisk DataBase
FireRiskDB
WeatherData Database
WeatherDataDB
None

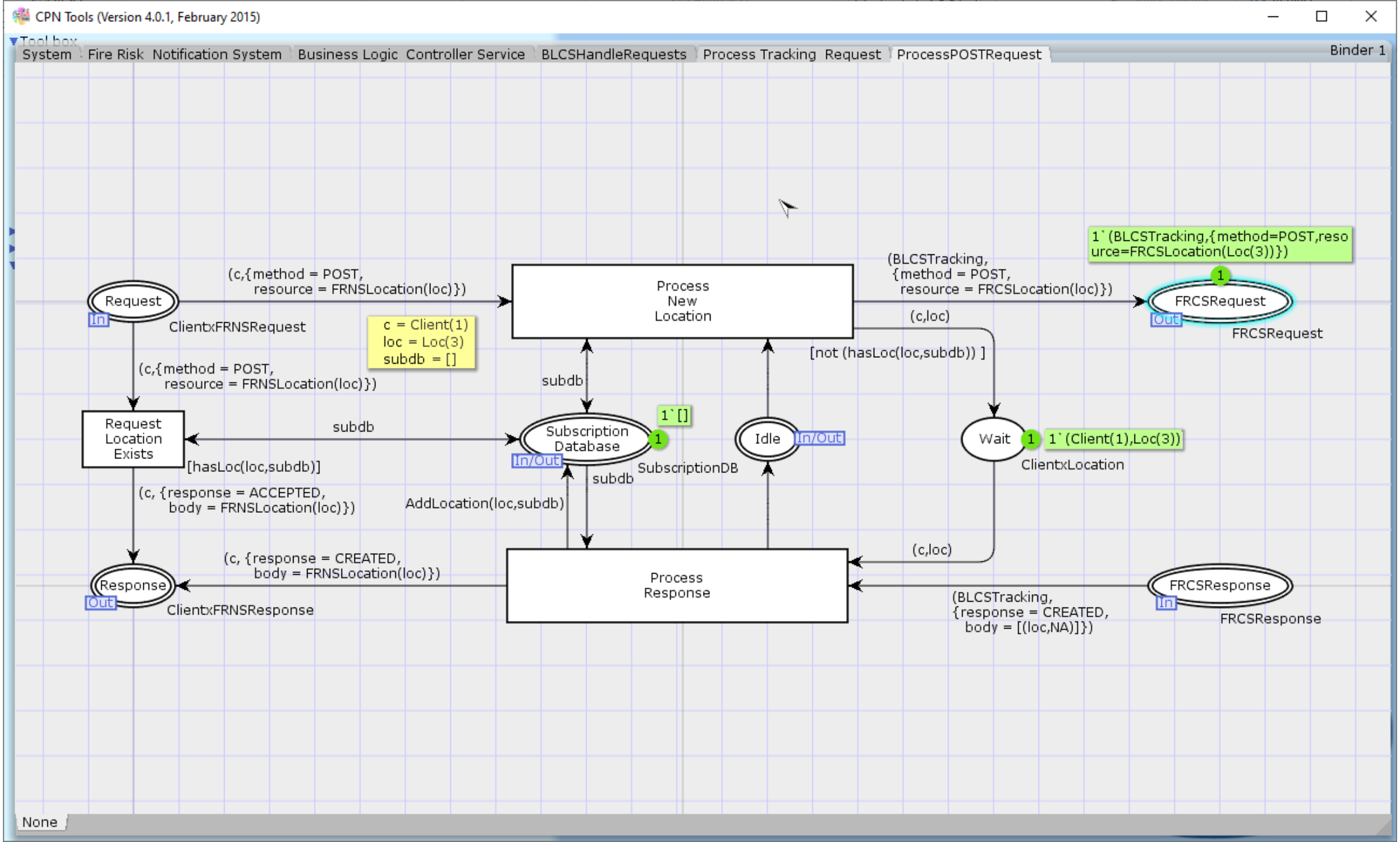

CPN Tools (Version 4.0.1, February 2015)
Tool box
System
Fire Risk Notification System
Business Logic Controller Service
Binder 1
1`(Client(1),{method=POST,resource =FRNSLocation(Loc(3))})
Request
1
In
ClientxFRNSRequest
BLCSHandleRequests
Response
Out
ClientxFRNSResponse
1`()
1
Idle
1
Subscription Database
1`[]
1
In/Out
SubscriptionDB
FRCSResponse
In
FRCSResponse
FRCSRequest
Out
FRCSRequest
Notification
Out
ClientxFRNSNotification
Scheduled Notifications
Scheduled Update
None

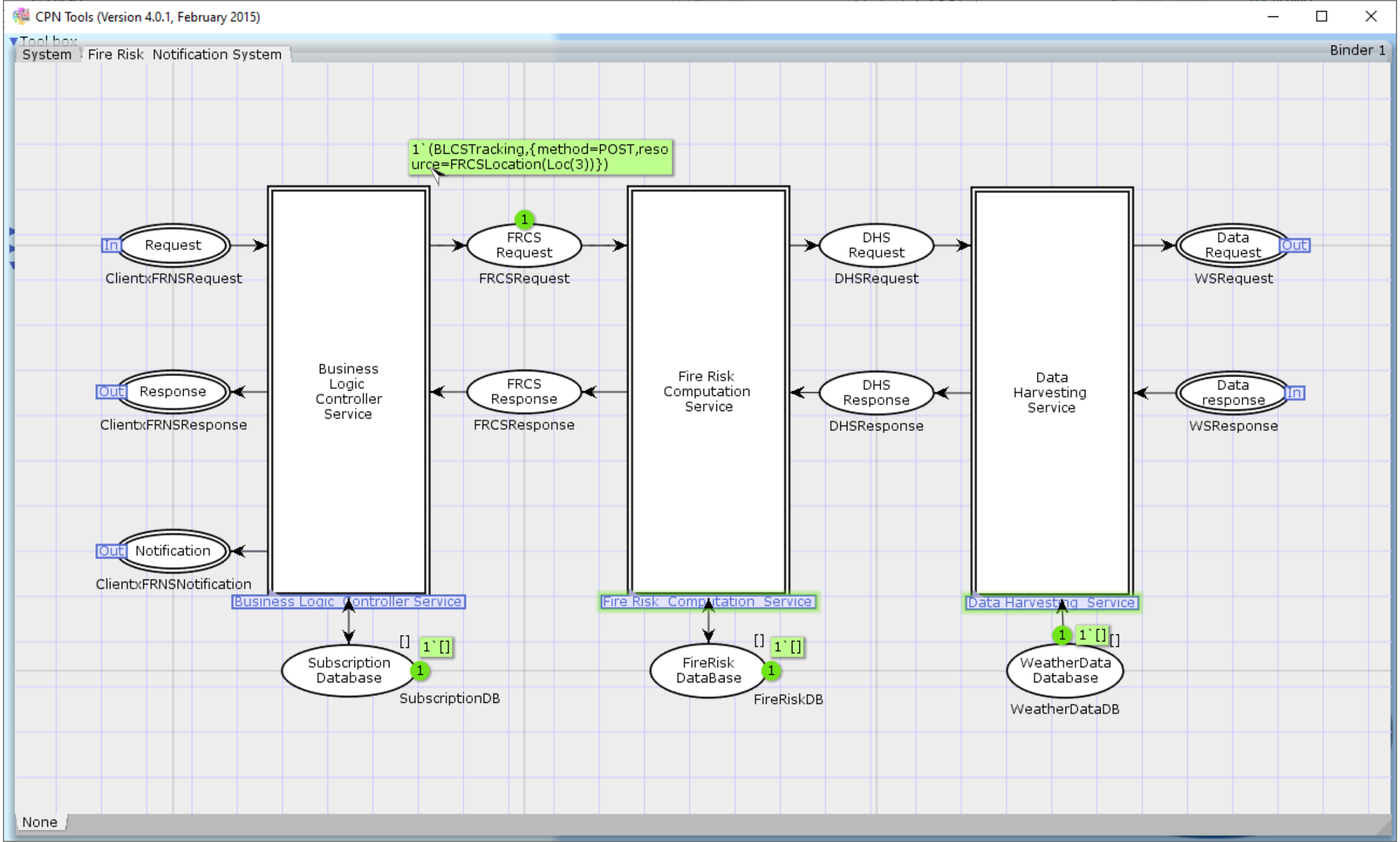

CPN Tools (Version 4.0.1, February 2015)
Tool box
System
Fire Risk Notification System
Business Logic Controller Service
BLCSHandleRequests
Binder 1
Process Tracking Request
Process Tracking Request
In/Out
1
1`()
Process Subscription Request
Process Subscription Request
Subscription Database
1
1`[]
In/Out
SubscriptionDB
Process Firerisk Request
Process Firerisk Request
Request
1
In
ClientxFRNSRequest
1`(Client(1),{method=POST,resource =FRNSLocation(Loc(3))})
Response
Out
ClientxFRNSResponse
FRCSRequest
Out
FRCSRequest
FRCSResponse
In
FRCSResponse
None

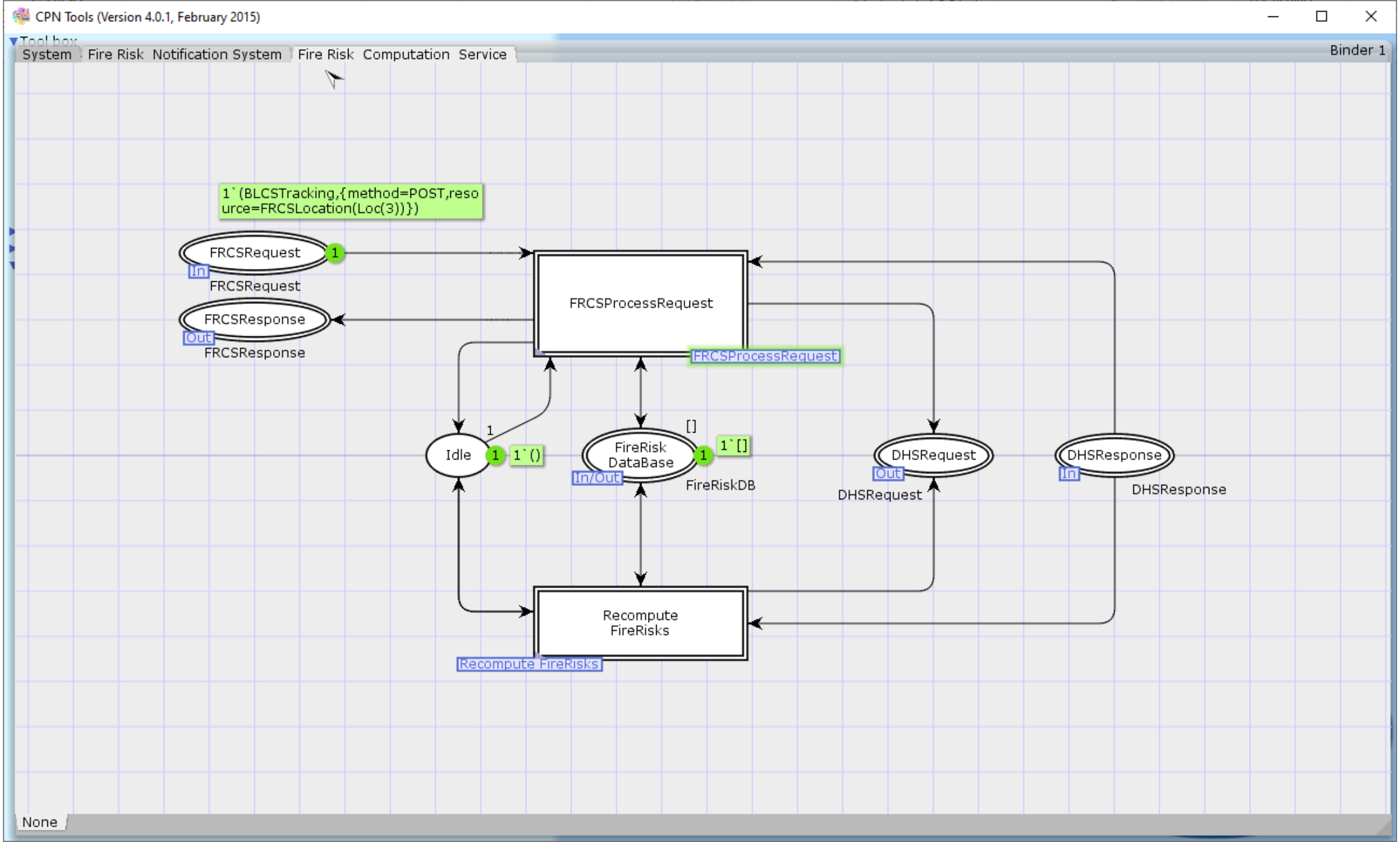

CPN Tools (Version 4.0.1, February 2015)
Tool box
System
Fire Risk Notification System
Business Logic Controller Service
BLCSHandleRequests
Process Tracking Request
Binder 1
Process POST Request
ProcessPOSTRequest
1`(Client(1),{method=POST,resource =FRNSLocation(Loc(3))})
Request
In
ClientxFRNSRequest
Response
Out
ClientxFRNSResponse
1`()
In/Out
1`[]
Subscription Database
In/Out
SubscriptionDB
FRCSRequest
Out
FRCSRequest
FRCSResponse
In
FRCSResponse
Process DELETE Request
ProcessDELETERequest
None

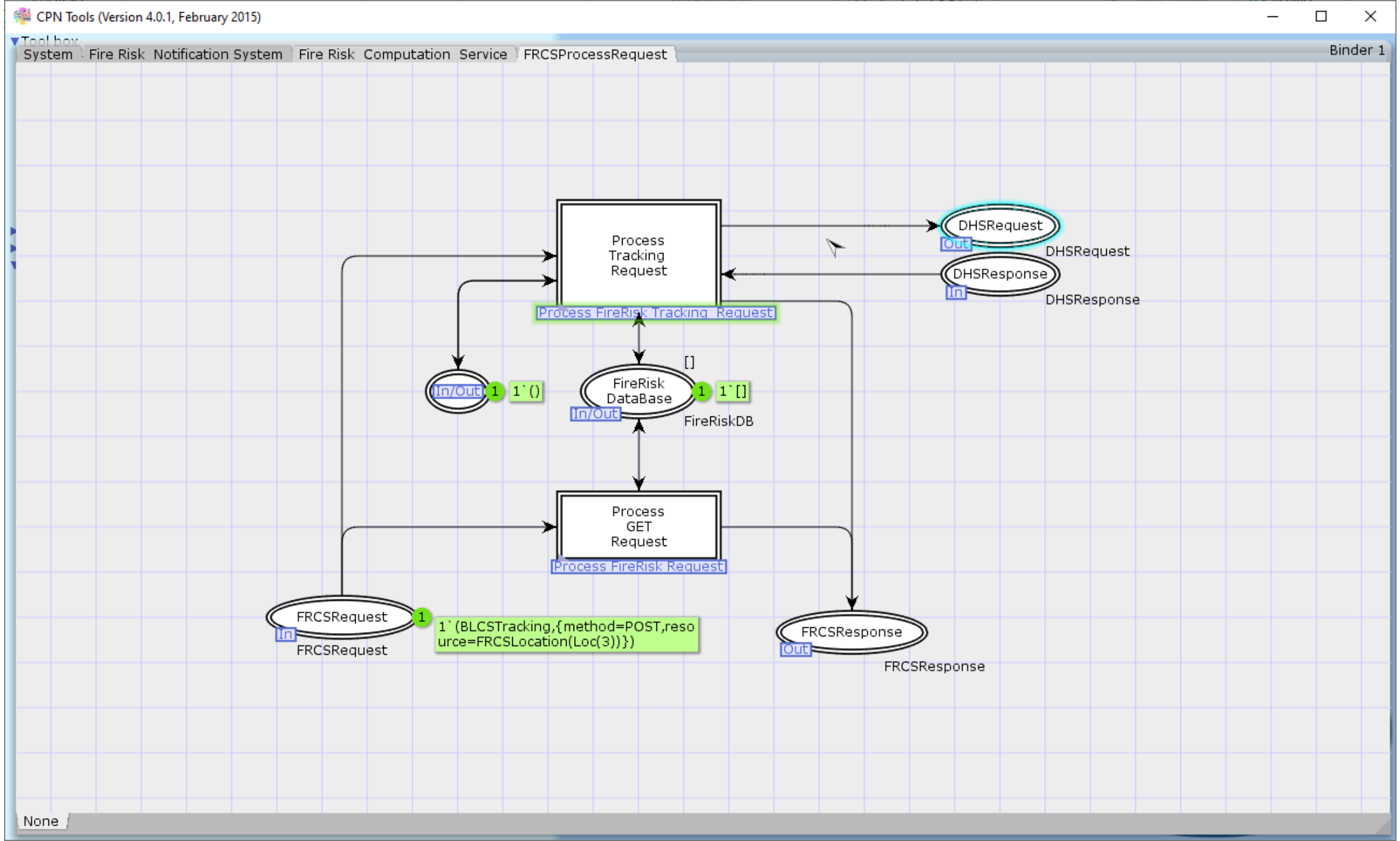

CPN Tools (Version 4.0.1, February 2015)
Tool box
System
Fire Risk Notification System
Business Logic Controller Service
BLCSHandleRequests
Process Tracking Request
ProcessPOSTRequest
Binder 1
1`(Client(1),{method=POST,resource =FRNSLocation(Loc(3))})
Request
In
ClientxFRNSRequest
(c,{method = POST, resource = FRNSLocation(loc)})
Process New Location
(BLCSTracking, {method = POST, resource = FRCSLocation(loc)})
FRCSRequest
Out
FRCSRequest
(c,loc)
[not (hasLoc(loc,subdb))]
subdb
1`[]
1`()
Request Location Exists
subdb
Subscription Database
In/Out
SubscriptionDB
Idle
In/Out
Wait
ClientxLocation
[hasLoc(loc,subdb)]
(c, {response = ACCEPTED, body = FRNSLocation(loc)})
AddLocation(loc,subdb)
subdb
(c,loc)
(c, {response = CREATED, body = FRNSLocation(loc)})
Response
Out
ClientxFRNSResponse
Process Response
FRCSResponse
In
FRCSResponse
(BLCSTracking, {response = CREATED, body = [(loc,NA)]})
None

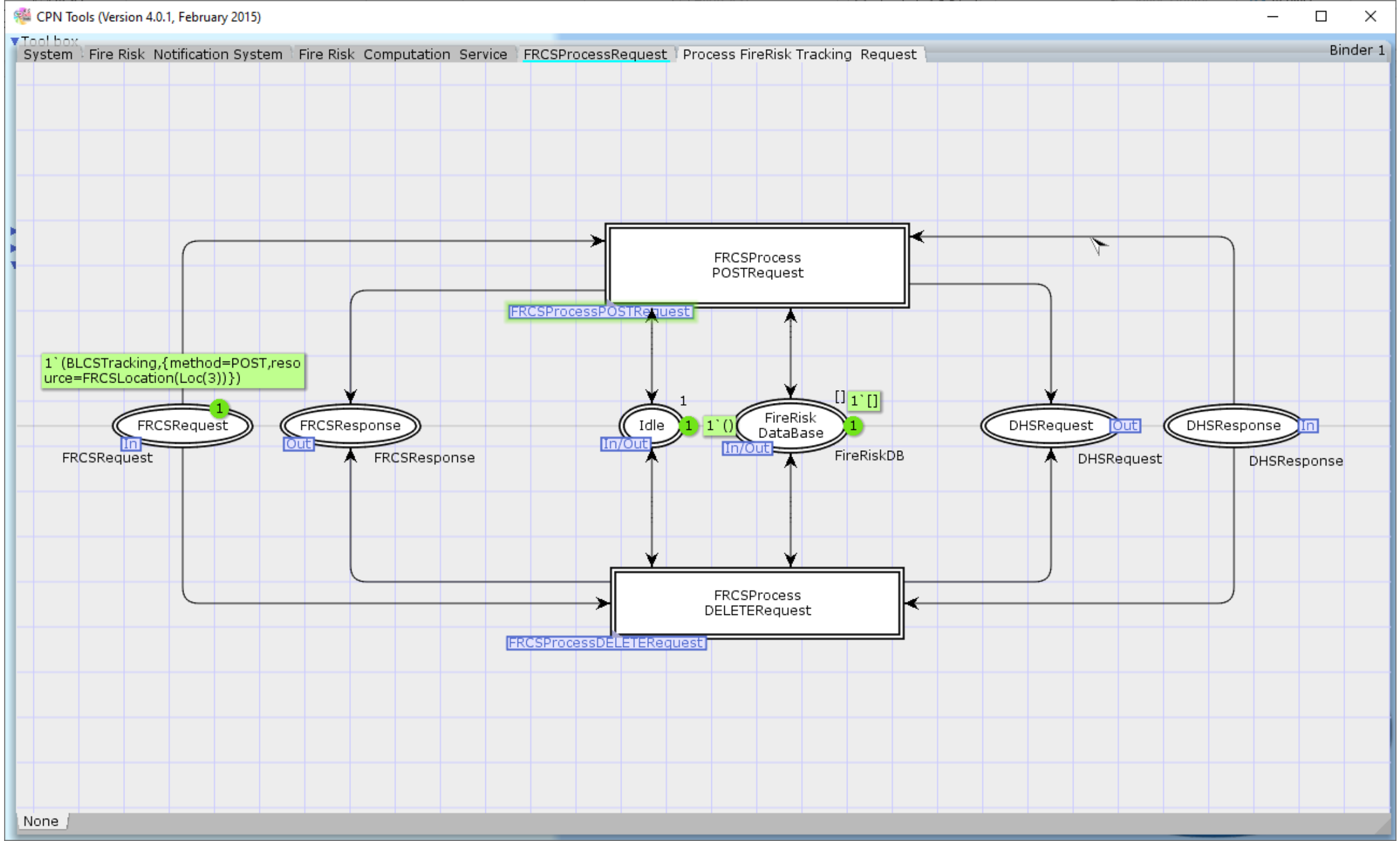

CPN Tools (Version 4.0.1, February 2015)
Tool box
System
Fire Risk Notification System
Business Logic Controller Service
BLCSHandleRequests
Process Tracking Request
ProcessPOSTRequest
Binder 1
1`(Client(1),{method=POST,resource =FRNSLocation(Loc(3))})
Request
In
ClientxFRNSRequest
(c,{method = POST, resource = FRNSLocation(loc)})
c = Client(1)
loc = Loc(3)
subdb = []
Process New Location
(BLCSTracking, {method = POST, resource = FRCSLocation(loc)})
FRCSRequest
Out
FRCSRequest
(c,loc)
[not (hasLoc(loc,subdb))]
subdb
1`[]
1`()
Request Location Exists
subdb
Subscription Database
In/Out
SubscriptionDB
Idle
In/Out
Wait
ClientxLocation
[hasLoc(loc,subdb)]
(c, {response = ACCEPTED, body = FRNSLocation(loc)})
AddLocation(loc,subdb)
subdb
(c,loc)
(c, {response = CREATED, body = FRNSLocation(loc)})
Response
Out
ClientxFRNSResponse
Process Response
FRCSResponse
In
FRCSResponse
(BLCSTracking, {response = CREATED, body = [(loc,NA)]})
None

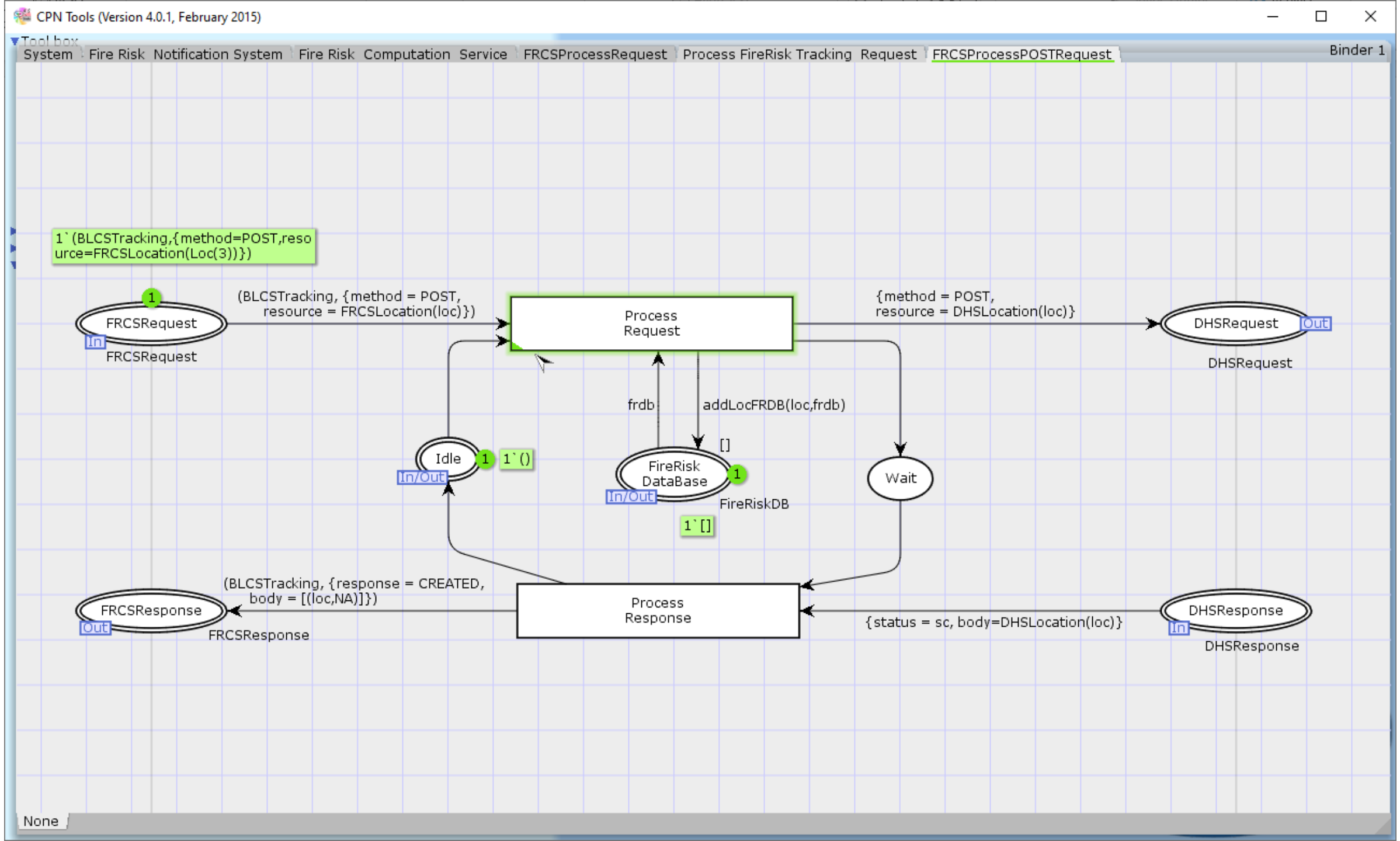

CPN Tools (Version 4.0.1, February 2015)
Tool box
System
Fire Risk Notification System
Business Logic Controller Service
BLCSHandleRequests
Process Tracking Request
ProcessPOSTRequest
Binder 1
Request
In
ClientxFRNSRequest
(c,{method = POST,
resource = FRNSLocation(loc)})
c = Client(1)
loc = Loc(3)
subdb = []
Process
New
Location
(BLCSTracking,
{method = POST,
resource = FRCSLocation(loc)})
1`(BLCSTracking,{method=POST,resource=FRCSLocation(Loc(3))})
FRCSRequest
Out
FRCSRequest
(c,loc)
[not (hasLoc(loc,subdb))]
(c,{method = POST,
resource = FRNSLocation(loc)})
subdb
Request
Location
Exists
[hasLoc(loc,subdb)]
subdb
Subscription
Database
1`[]
In/Out
SubscriptionDB
Idle
In/Out
Wait
1`(Client(1),Loc(3))
ClientxLocation
(c, {response = ACCEPTED,
body = FRNSLocation(loc)})
AddLocation(loc,subdb)
subdb
(c,loc)
(c, {response = CREATED,
body = FRNSLocation(loc)})
Response
Out
ClientxFRNSResponse
Process
Response
FRCSResponse
In
FRCSResponse
(BLCSTracking,
{response = CREATED,
body = [(loc,NA)]})
None

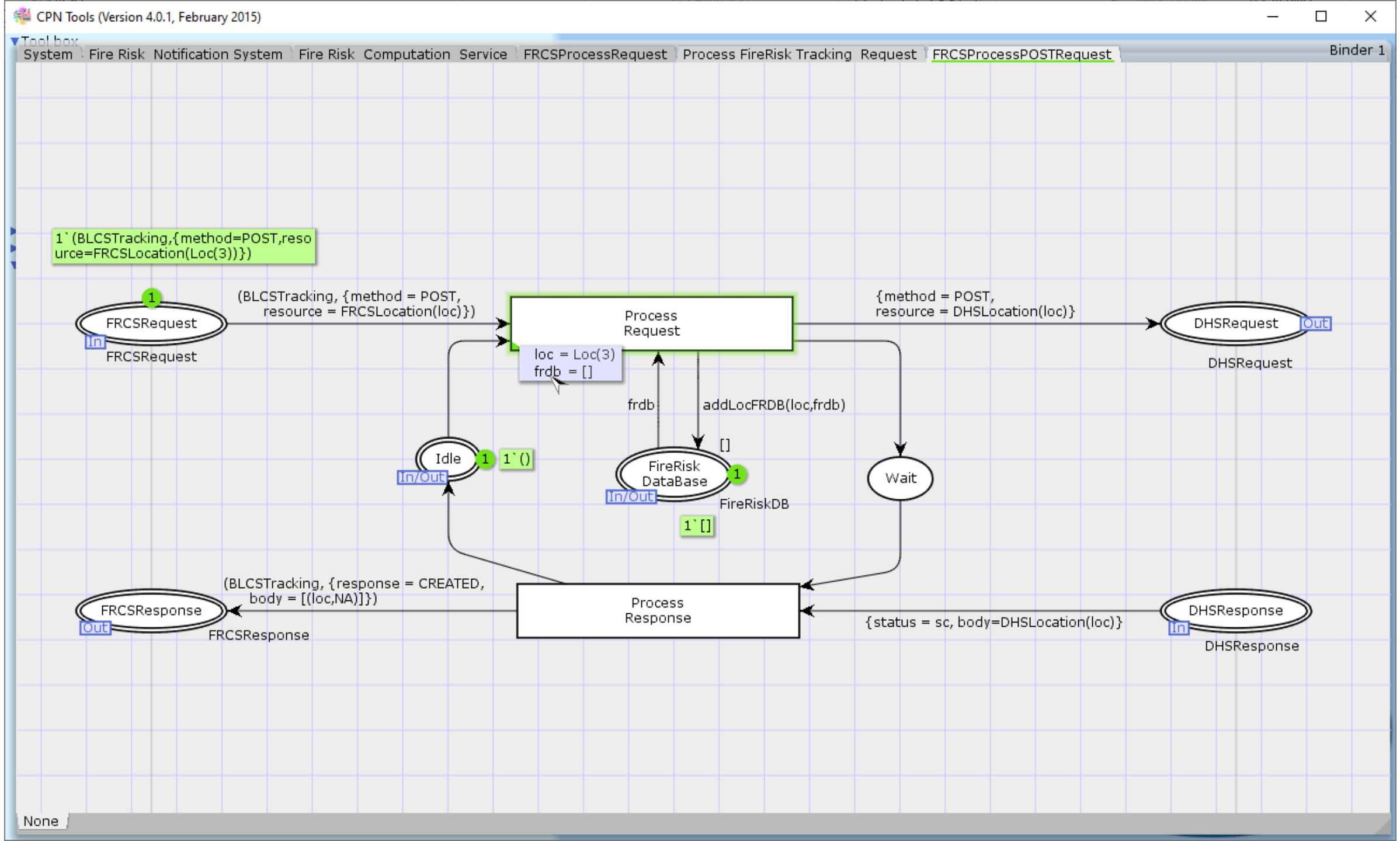

CPN Tools (Version 4.0.1, February 2015)
Tool box
System
Fire Risk Notification System
Binder 1
1`(BLCSTracking,{method=POST,resource=FRCSLocation(Loc(3))})
In
Request
ClientxFRNSRequest
1
FRCS Request
FRCSRequest
DHS Request
DHSRequest
Data Request
Out
WSRequest
Business Logic Controller Service
Fire Risk Computation Service
Data Harvesting Service
Out
Response
ClientxFRNSResponse
FRCS Response
FRCSResponse
DHS Response
DHSResponse
Data response
In
WSResponse
Out
Notification
ClientxFRNSNotification
Business Logic Controller Service
Fire Risk Computation Service
Data Harvesting Service
[]
1`[]
Subscription Database
1
SubscriptionDB
[]
1`[]
FireRisk DataBase
1
FireRiskDB
1
1`[]
[]
WeatherData Database
WeatherDataDB
None

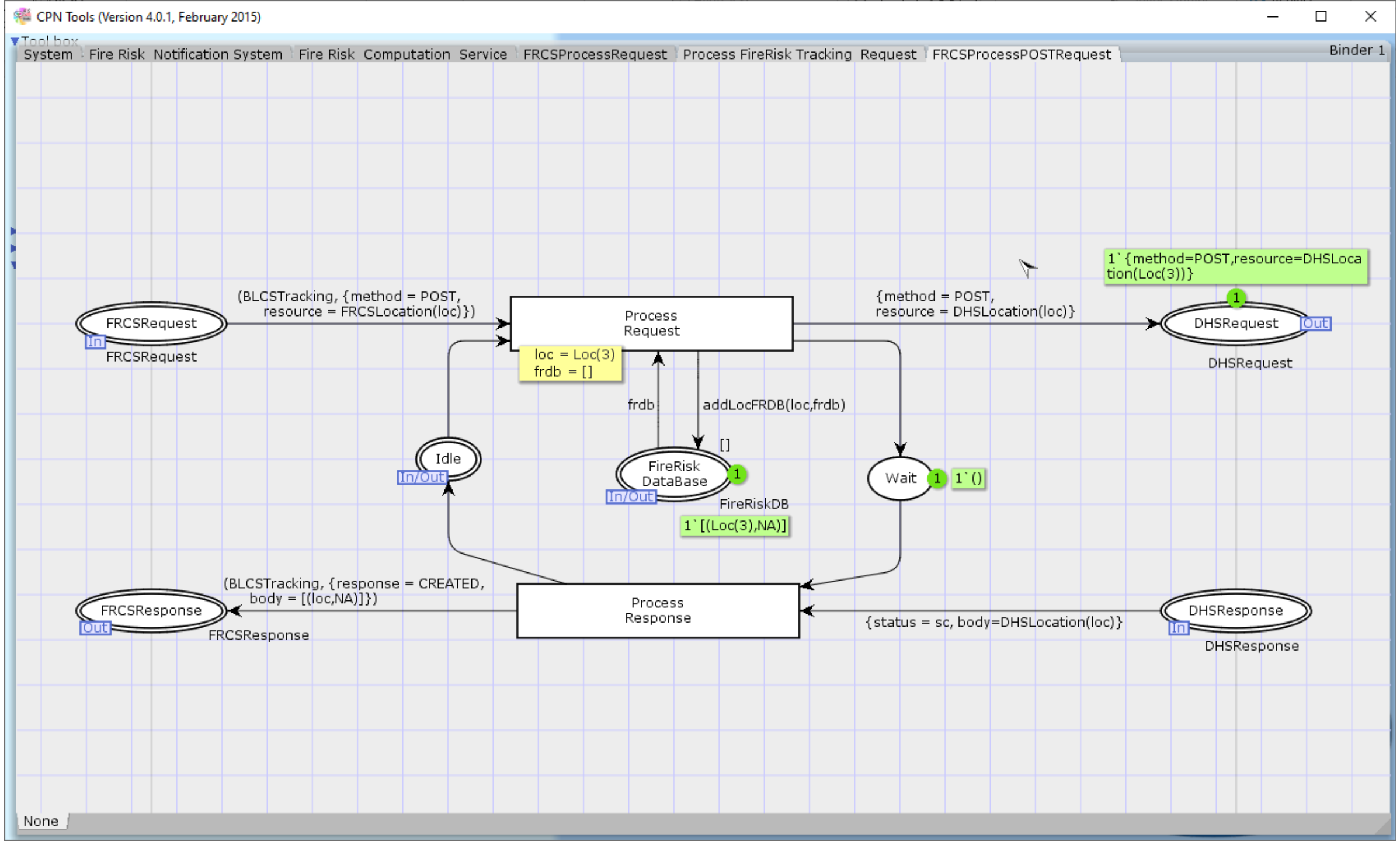

CPN Tools (Version 4.0.1, February 2015)
Tool box
System
Fire Risk Notification System
Fire Risk Computation Service
Binder 1
1`(BLCSTracking,{method=POST,resource=FRCSLocation(Loc(3))})
FRCSRequest
1
In
FRCSRequest
FRCSResponse
Out
FRCSResponse
FRCSProcessRequest
FRCSProcessRequest
1
Idle
1
1`()
[]
FireRisk DataBase
1
1`[]
In/Out
FireRiskDB
DHSRequest
Out
DHSRequest
DHSResponse
In
DHSResponse
Recompute FireRisks
Recompute FireRisks
None

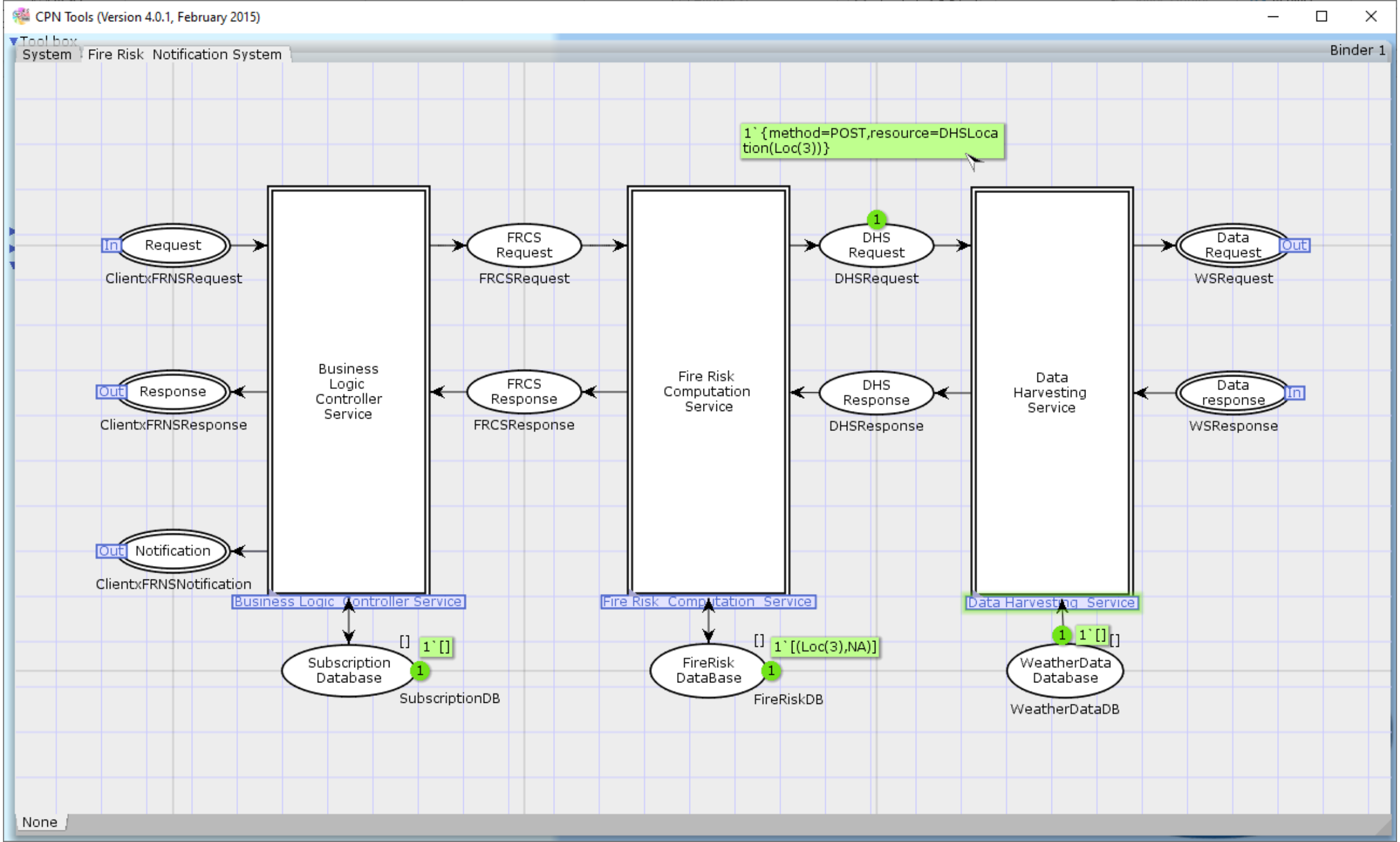

CPN Tools (Version 4.0.1, February 2015)
Tool box
System
Fire Risk Notification System
Fire Risk Computation Service
FRCSProcessRequest
Binder 1
Process Tracking Request
Process FireRisk Tracking Request
DHSRequest
Out
DHSRequest
DHSResponse
In
DHSResponse
In/Out
1
1`()
FireRisk DataBase
[]
1
1`[]
In/Out
FireRiskDB
Process GET Request
Process FireRisk Request
FRCSRequest
In
FRCSRequest
1
1`(BLCSTracking,{method=POST,resource=FRCSLocation(Loc(3))})
FRCSResponse
Out
FRCSResponse
None

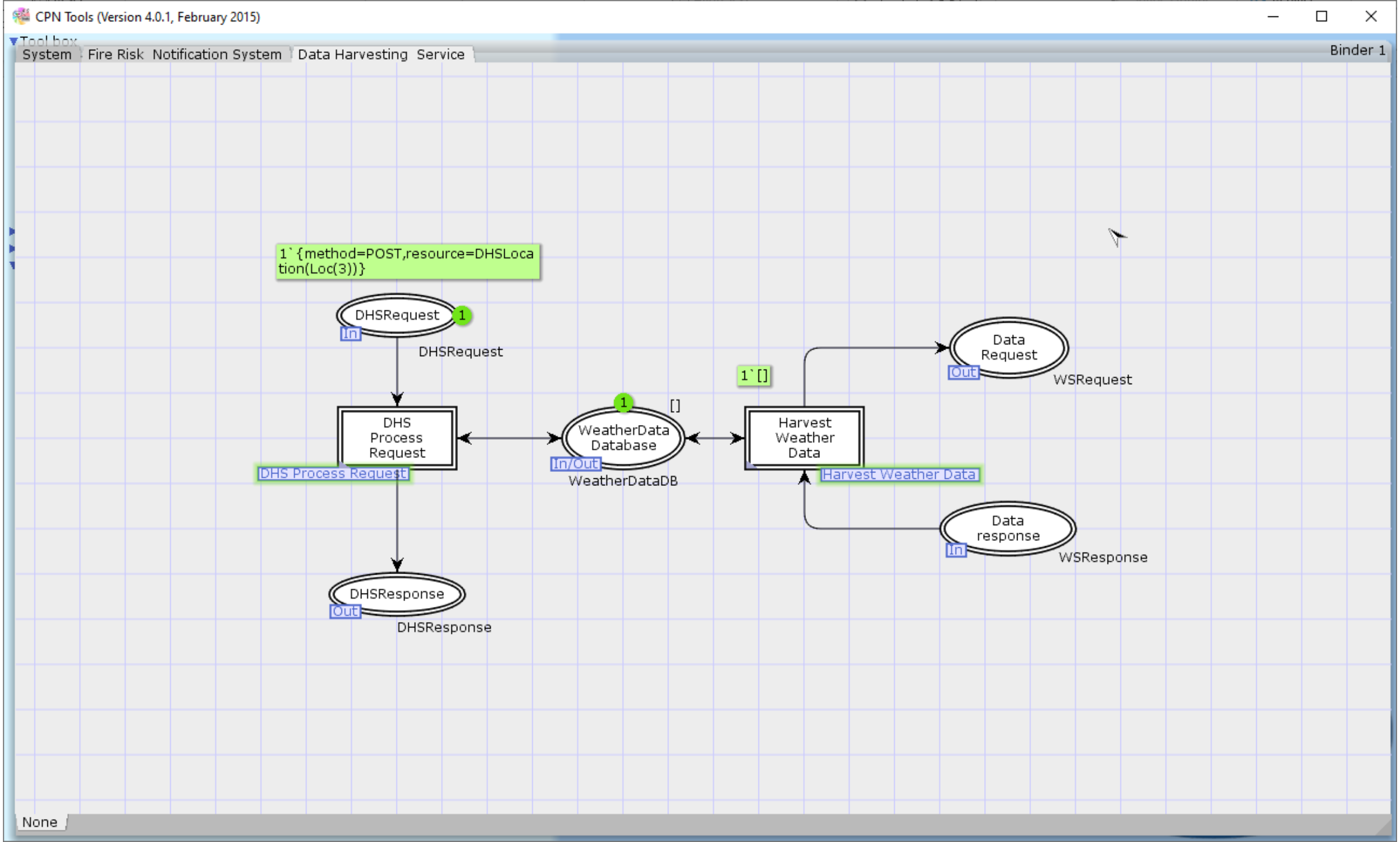

CPN Tools (Version 4.0.1, February 2015)
Tool box
System
Fire Risk Notification System
Fire Risk Computation Service
FRCSProcessRequest
Process FireRisk Tracking Request
Binder 1
FRCSProcess POSTRequest
FRCSProcessPOSTRequest
1`(BLCSTracking,{method=POST,resource=FRCSLocation(Loc(3))})
FRCSRequest
In
FRCSRequest
FRCSResponse
Out
FRCSResponse
Idle
1
In/Out
1`()
FireRisk DataBase
[]
1`[]
In/Out
FireRiskDB
DHSRequest
Out
DHSRequest
DHSResponse
In
DHSResponse
FRCSProcess DELETERequest
FRCSProcessDELETERequest
None

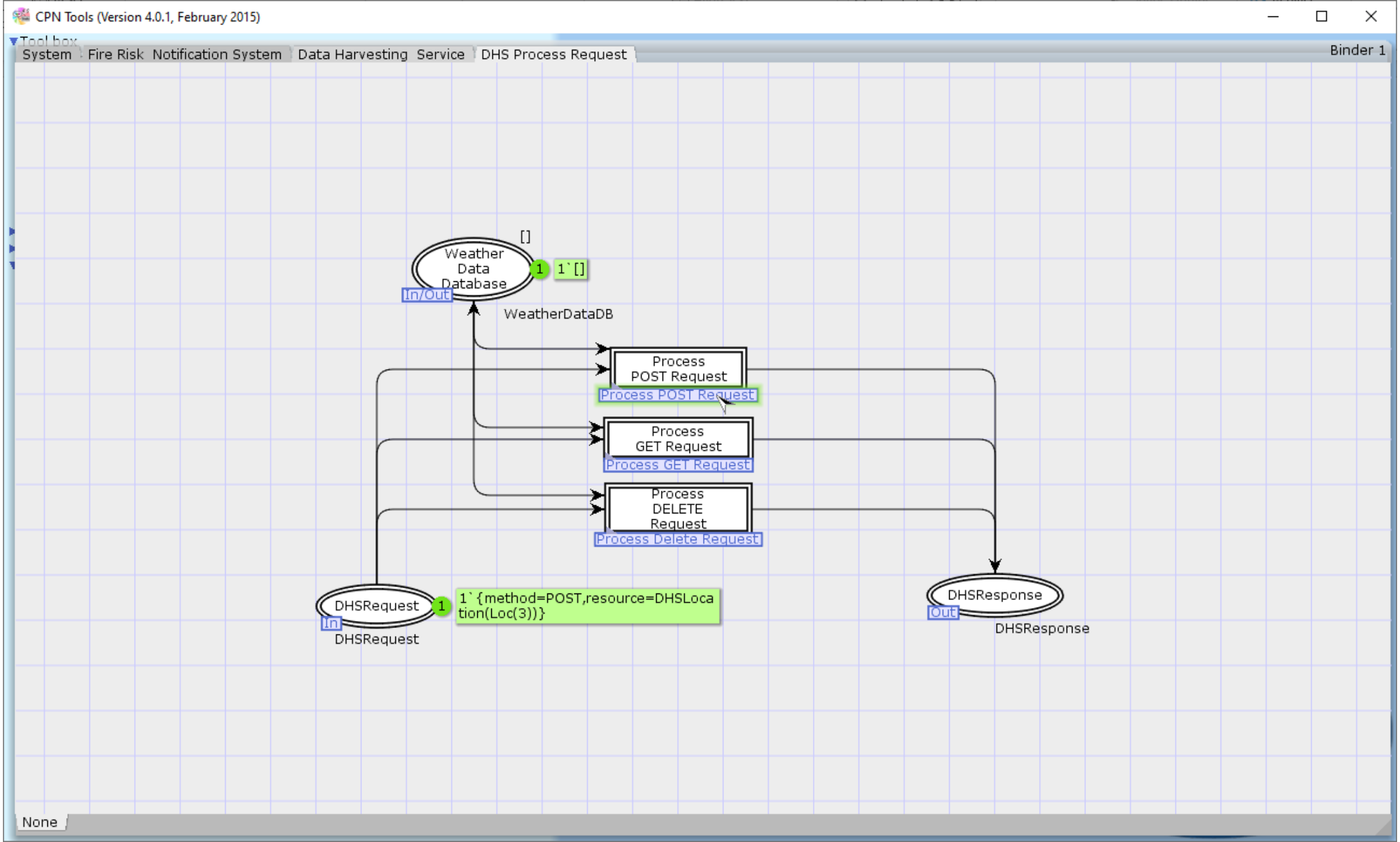

CPN Tools (Version 4.0.1, February 2015)
Tool box
System
Fire Risk Notification System
Fire Risk Computation Service
FRCSProcessRequest
Process FireRisk Tracking Request
FRCSProcessPOSTRequest
Binder 1
1`(BLCSTracking,{method=POST,resource=FRCSLocation(Loc(3))})
FRCSRequest
In
FRCSRequest
(BLCSTracking, {method = POST, resource = FRCSLocation(loc)})
Process Request
{method = POST, resource = DHSLocation(loc)}
DHSRequest
Out
DHSRequest
frdb
addLocFRDB(loc,frdb)
[]
Idle
1
1`()
In/Out
FireRisk DataBase
1
In/Out
FireRiskDB
1`[]
Wait
(BLCSTracking, {response = CREATED, body = [(loc,NA)]})
FRCSResponse
Out
FRCSResponse
Process Response
{status = sc, body=DHSLocation(loc)}
DHSResponse
In
DHSResponse
None

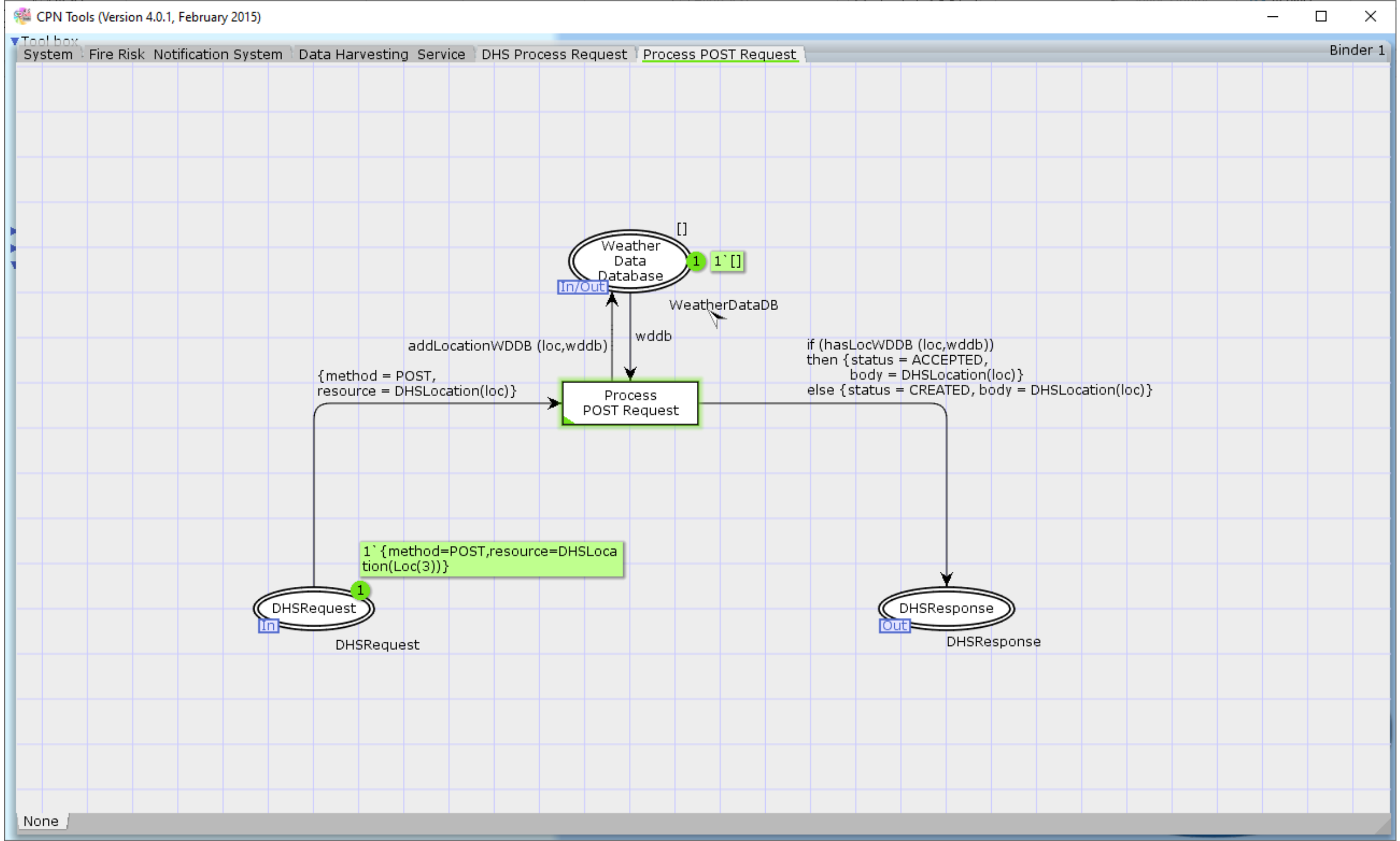

CPN Tools (Version 4.0.1, February 2015)
Tool box
System
Fire Risk Notification System
Fire Risk Computation Service
FRCSProcessRequest
Process FireRisk Tracking Request
FRCSProcessPOSTRequest
Binder 1
1`(BLCSTracking,{method=POST,resource=FRCSLocation(Loc(3))})
FRCSRequest
In
FRCSRequest
(BLCSTracking, {method = POST, resource = FRCSLocation(loc)})
Process Request
loc = Loc(3)
frdb = []
{method = POST, resource = DHSLocation(loc)}
DHSRequest
Out
DHSRequest
frdb
addLocFRDB(loc,frdb)
[]
Idle
1
1`()
In/Out
FireRisk DataBase
1
In/Out
FireRiskDB
1`[]
Wait
(BLCSTracking, {response = CREATED, body = [(loc,NA)]})
FRCSResponse
Out
FRCSResponse
Process Response
{status = sc, body=DHSLocation(loc)}
DHSResponse
In
DHSResponse
None

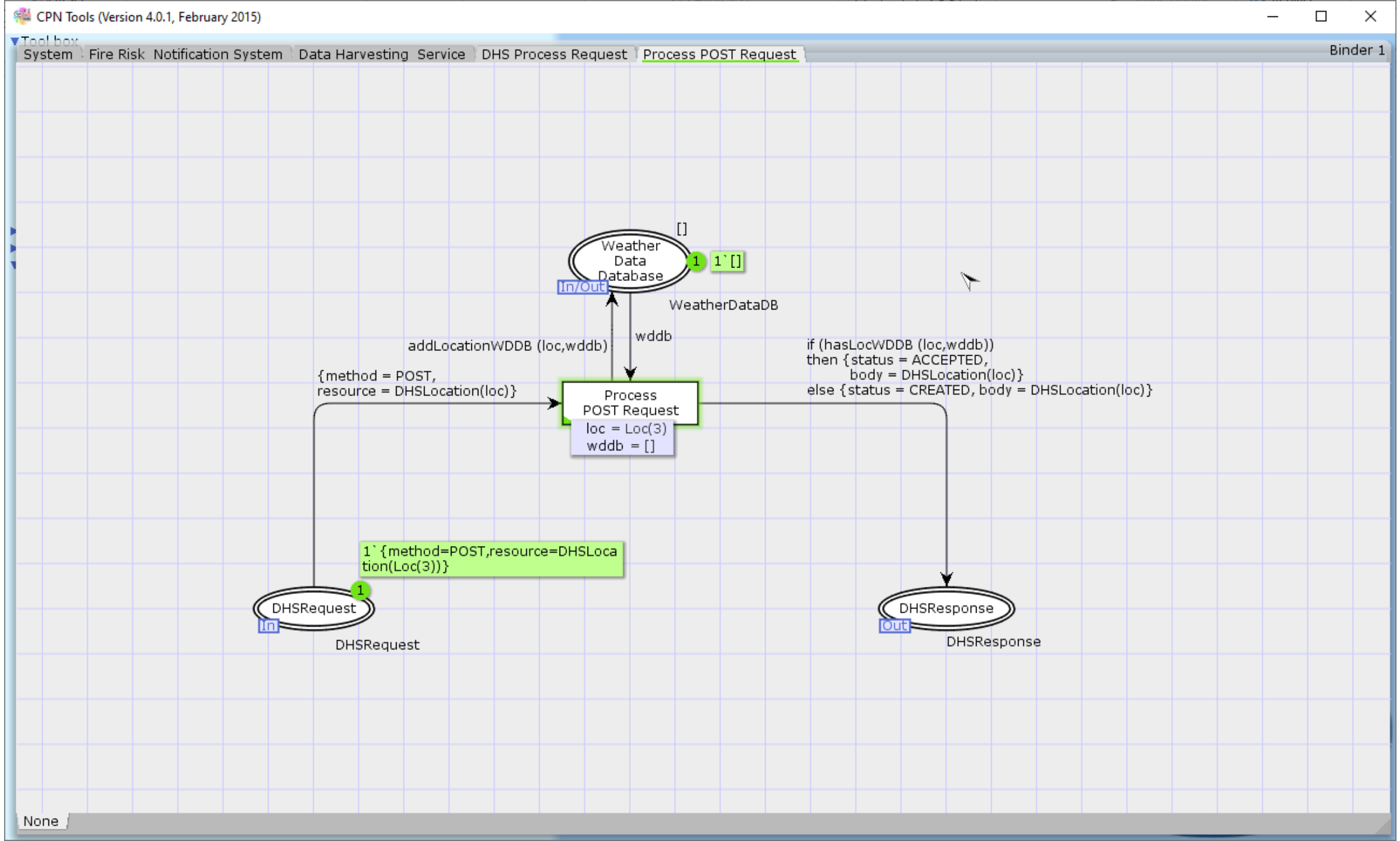

CPN Tools (Version 4.0.1, February 2015)
Tool box
System
Fire Risk Notification System
Fire Risk Computation Service
FRCSProcessRequest
Process FireRisk Tracking Request
FRCSProcessPOSTRequest
Binder 1
FRCSRequest
In
FRCSRequest
(BLCSTracking, {method = POST,
resource = FRCSLocation(loc)})
Process
Request
loc = Loc(3)
frdb = []
{method = POST,
resource = DHSLocation(loc)}
1`{method=POST,resource=DHSLocation(Loc(3))}
1
DHSRequest
Out
DHSRequest
frdb
addLocFRDB(loc,frdb)
[]
Idle
In/Out
FireRisk
DataBase
1
In/Out
FireRiskDB
1`[(Loc(3),NA)]
Wait
1
1`()
(BLCSTracking, {response = CREATED,
body = [(loc,NA)]})
FRCSResponse
Out
FRCSResponse
Process
Response
{status = sc, body=DHSLocation(loc)}
DHSResponse
In
DHSResponse
None

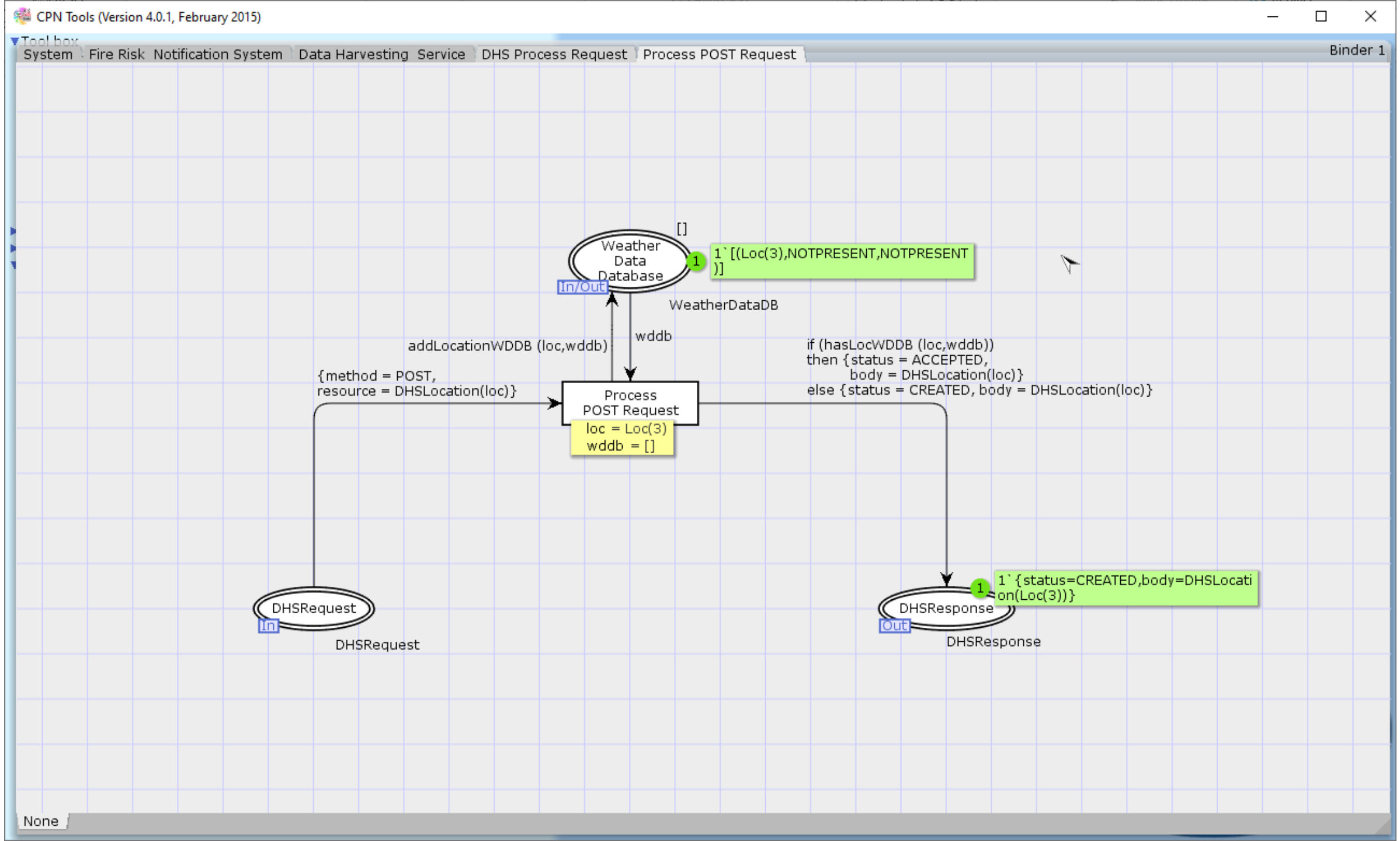

CPN Tools (Version 4.0.1, February 2015)
Tool box
System
Fire Risk Notification System
Binder 1
1`{method=POST,resource=DHSLocation(Loc(3))}
In
Request
ClientxFRNSRequest
Business Logic Controller Service
FRCS Request
FRCSRequest
Fire Risk Computation Service
1
DHS Request
DHSRequest
Data Harvesting Service
Data Request
Out
WSRequest
Out
Response
ClientxFRNSResponse
FRCS Response
FRCSResponse
DHS Response
DHSResponse
Data response
In
WSResponse
Out
Notification
ClientxFRNSNotification
Business Logic Controller Service
Fire Risk Computation Service
Data Harvesting Service
[]
1`[]
Subscription Database
1
SubscriptionDB
[]
1`[(Loc(3),NA)]
FireRisk DataBase
1
FireRiskDB
1
1`[]
[]
WeatherData Database
WeatherDataDB
None

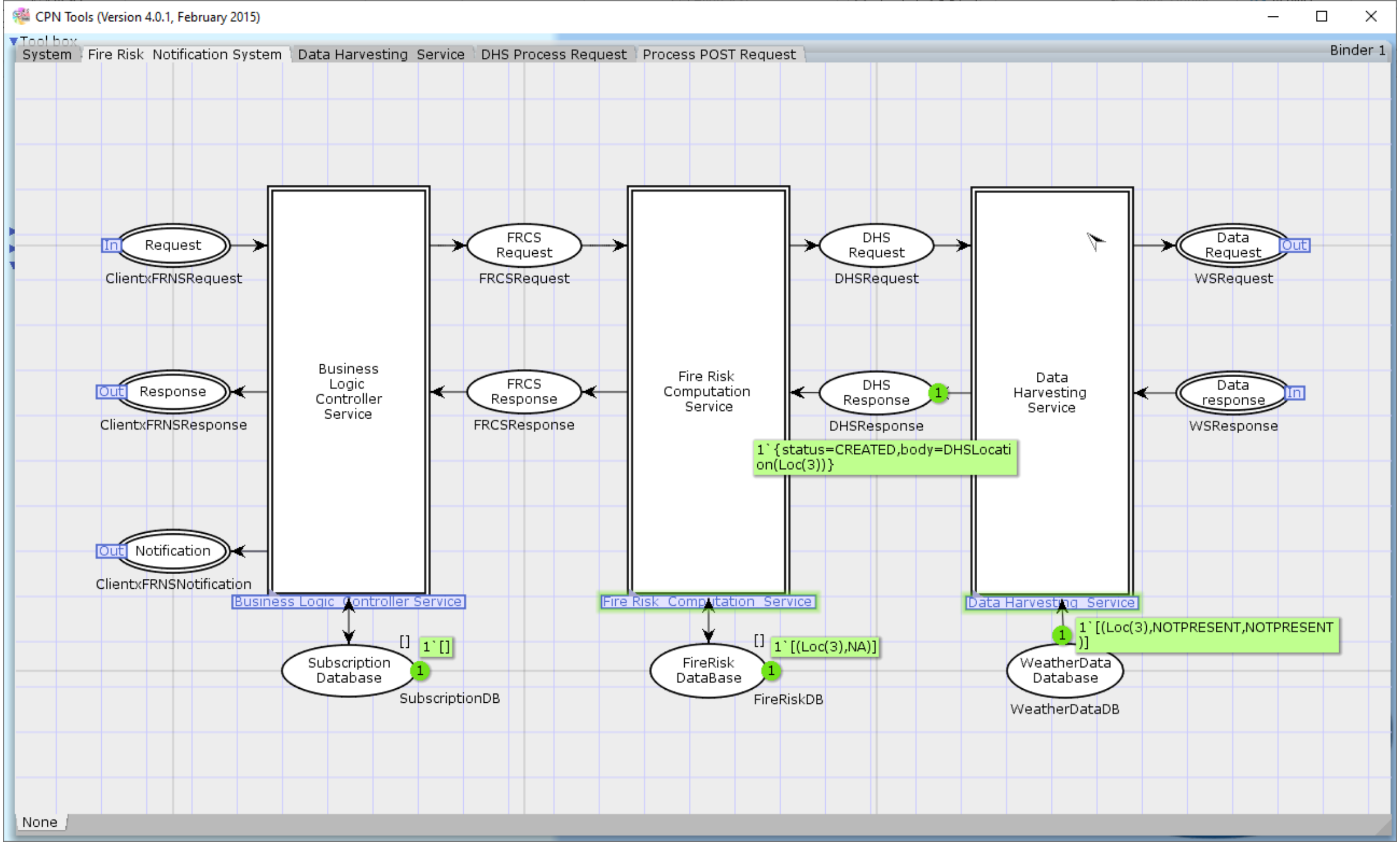

CPN Tools (Version 4.0.1, February 2015)
Tool box
System
Fire Risk Notification System
Data Harvesting Service
Binder 1
1`{method=POST,resource=DHSLocation(Loc(3))}
DHSRequest
1
In
DHSRequest
DHS Process Request
DHS Process Request
DHSResponse
Out
DHSResponse
1
WeatherData Database
In/Out
WeatherDataDB
[]
1`[]
Harvest Weather Data
Harvest Weather Data
Data Request
Out
WSRequest
Data response
In
WSResponse
None


CPN Tools (Version 4.0.1, February 2015)
Tool box
System
Fire Risk Notification System
Data Harvesting Service
DHS Process Request
Binder 1
[]
Weather Data Database
1
1`[]
In/Out
WeatherDataDB
Process POST Request
Process POST Request
Process GET Request
Process GET Request
Process DELETE Request
Process Delete Request
DHSRequest
1
1`{method=POST,resource=DHSLocation(Loc(3))}
In
DHSRequest
DHSResponse
Out
DHSResponse
None


CPN Tools (Version 4.0.1, February 2015)
Tool box
System
Fire Risk Notification System
Data Harvesting Service
DHS Process Request
Process POST Request
Binder 1
[]
Weather Data Database
1
1`[]
In/Out
WeatherDataDB
wddb
addLocationWDDB (loc,wddb)
{method = POST,
resource = DHSLocation(loc)}
Process POST Request
if (hasLocWDDB (loc,wddb))
then {status = ACCEPTED,
body = DHSLocation(loc)}
else {status = CREATED, body = DHSLocation(loc)}
1`{method=POST,resource=DHSLocation(Loc(3))}
1
DHSRequest
In
DHSRequest
DHSResponse
Out
DHSResponse
None


CPN Tools (Version 4.0.1, February 2015)
Tool box
System
Fire Risk Notification System
Data Harvesting Service
DHS Process Request
Process POST Request
Binder 1
[]
Weather Data Database
1
1`[]
In/Out
WeatherDataDB
wddb
addLocationWDDB (loc,wddb)
if (hasLocWDDB (loc,wddb))
then {status = ACCEPTED,
body = DHSLocation(loc)}
else {status = CREATED, body = DHSLocation(loc)}
{method = POST,
resource = DHSLocation(loc)}
Process POST Request
loc = Loc(3)
wddb = []
1`{method=POST,resource=DHSLocation(Loc(3))}
1
DHSRequest
In
DHSRequest
DHSResponse
Out
DHSResponse
None


CPN Tools (Version 4.0.1, February 2015)
Tool box
System
Fire Risk Notification System
Data Harvesting Service
DHS Process Request
Process POST Request
Binder 1
[]
Weather Data Database
1`[(Loc(3),NOTPRESENT,NOTPRESENT)]
In/Out
WeatherDataDB
addLocationWDDB (loc,wddb)
wddb
{method = POST, resource = DHSLocation(loc)}
Process POST Request
loc = Loc(3)
wddb = []
if (hasLocWDDB (loc,wddb)) then {status = ACCEPTED, body = DHSLocation(loc)} else {status = CREATED, body = DHSLocation(loc)}
DHSRequest
In
DHSRequest
1`{status=CREATED,body=DHSLocation(Loc(3))}
DHSResponse
Out
DHSResponse
None


CPN Tools (Version 4.0.1, February 2015)
Tool box
System
Fire Risk Notification System
Data Harvesting Service
DHS Process Request
Process POST Request
Binder 1
In
Request
ClientxFRNSRequest
FRCS Request
FRCSRequest
DHS Request
DHSRequest
Data Request
Out
WSRequest
Out
Response
ClientxFRNSResponse
Business Logic Controller Service
FRCS Response
FRCSResponse
Fire Risk Computation Service
DHS Response
1
DHSResponse
Data Harvesting Service
Data response
In
WSResponse
1`{status=CREATED,body=DHSLocation(Loc(3))}
Out
Notification
ClientxFRNSNotification
Business Logic Controller Service
Fire Risk Computation Service
Data Harvesting Service
[]
1`[]
Subscription Database
1
SubscriptionDB
[]
1`[(Loc(3),NA)]
FireRisk DataBase
1
FireRiskDB
1
1`[(Loc(3),NOTPRESENT,NOTPRESENT)]
WeatherData Database
WeatherDataDB
None

Model Validation

Objective

- Confirm basic behaviour

Method

- Single-step execution combined with interactive and automatic simulation (during and post building the model)

Findings

- Smaller modelling errors, corrected continously

Verification

Objective:

- Verify key behavioural properties

Method:

- State space exploration in conjunction with ASK-CTL library
  - Incremental verification approach, similar to [26]
    - Gradual verification of services
  - Verification of system-wide properties

[26] A. Rodríguez, L. M. Kristensen, A. Rutle, Formal modelling and incremental verification of the mqtt iot protocol, Trans. Petri Nets Other Model. Concurr. 14 (2019) 126–145.

Verification

Incremental process

1. Tracking
2. Subscriptions
3. Fire risk computations
4. Data harvesting
5. Inter-service properties (system wide)

Verification – Tracking properties (1)

T-P1 $\forall\, c \in C, l \in L : \boldsymbol{AGEF}\ (c \text{ request tracking of } l)$

It is always possible for any client to initiate the tracking of any location.

Consider the binding of the *Request Tracking* transition in the *Client* module.

T-P2 $\forall\, c \in C, l \in L : \boldsymbol{AG}\ ((c \text{ request tracking of } l) \Rightarrow \boldsymbol{EF}\ (l \text{ is being tracked}))$

If client request tracking of location, a future state exist where the location is being tracked.

Consider the databases of the three micro-services.

T-P3…

T-P4…

Other verification properties

2. Subscriptions
   S-P1 – S-P6
3. Fire risk computations
   F-P1 – F-P3
4. Data harvesting
   W-P1 – W-P2
5. Inter-service (system wide)
   A-P1 and **A-P2**

Properties relate to the purpose of the functions/services

DYNAMIC

Verification – Inter-service properties (5)

Properties which rely on collaboration between all services

A-P1 $\boldsymbol{AG}$ (no pending client requests $\Rightarrow$ consistent databases)

If no request, then databases are consistent.

Consider marking of the three places (databases)

A-P2 $\forall\, c \in C, l \in L$: $\boldsymbol{AG}$ (c is subscribed to l) $\Rightarrow$
$\boldsymbol{EF}$ (c recieves fire risk notification for l)

If a client is subscribe to a location, it is possible to receive a notification

Consider subscription database and occurrence of *Receive notification* transition

Results

State-space and verification statistics for configurations considered in the verification

Configuration	States	Arcs	G-Time	V-Time
C1-L1	3,435	11,901	< 1 s	< 1 s
C1-L2	215,181	739,797	2,093 s	555 s
C2-L1	274,581	1,238,395	9,100 s	1,404 s
C2-L2-*	14,556	62,535	27.7 s	26.6 s
C2-L3-*-Request	5,151	21,138	4 s	10.5 s
C2-L3-*-Notify	216	687	< 1 s	< 1 s
C3-L2-*-Request	18,654	91,596	49.5 s	67.0 s
C3-L2-*-Notify	372	1,311	< 1 s	< 1 s
C3-L3-*-Request	54,894	276,378	480.9 s	339.8 s
C3-L3-*-Notify	372	1,311	< 1 s	< 1 s

Conclusion

- Formal specification of the micro-service based architechture for the fire risk notification system
- General approach to modelling REST APIs using CPNs
- The CPN model within this paper will serve as basis for the final system implementation

Limitations and Future work

- Limited number of system configurations
- Investigate use of fairness assumptions
- Investigate CPN model test case generation



DYNAMIC