NetGuardians

RiskTech 100 2018

Gartner 2015 CoolVendor

Data Collection Framework

Ljupce Nikolov
September 2019

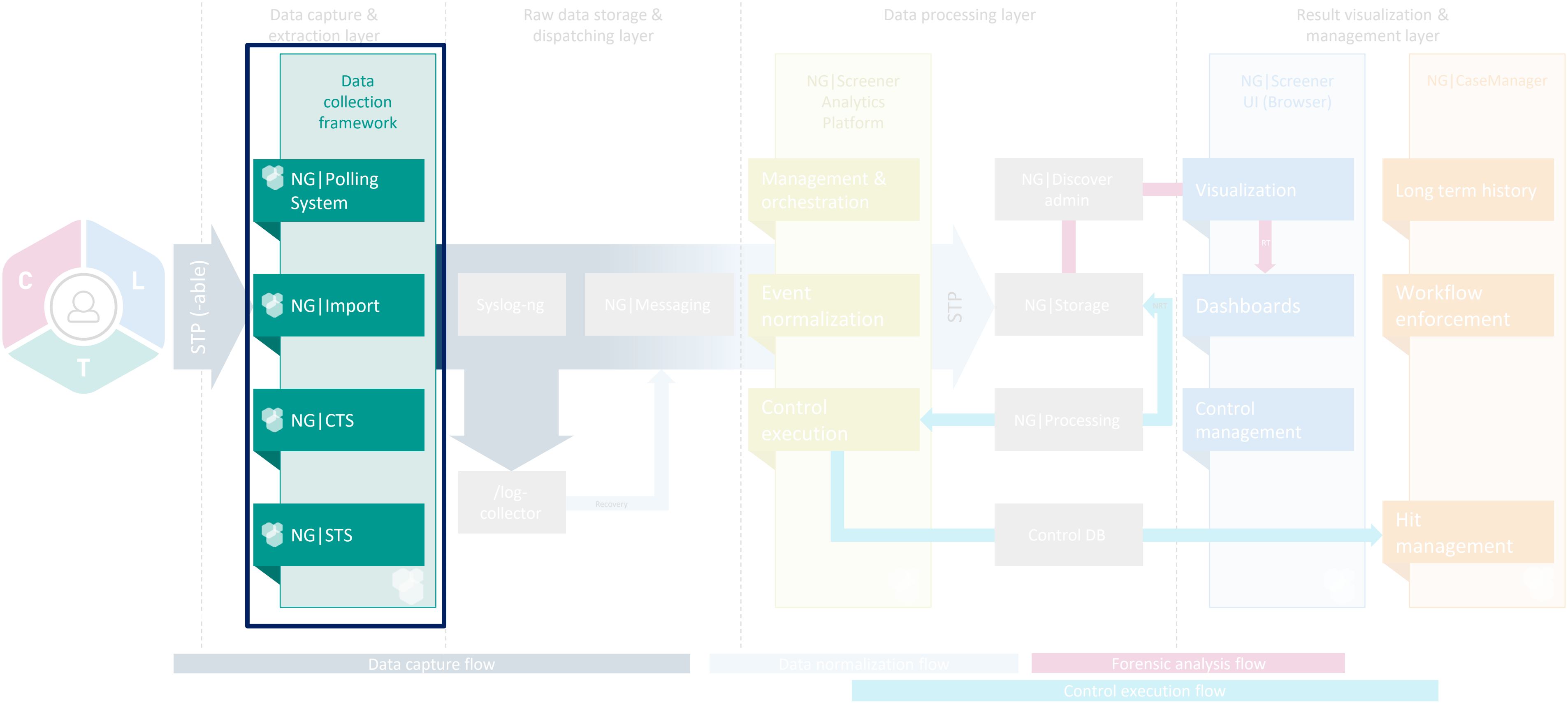
NG | Academy

swiss made software

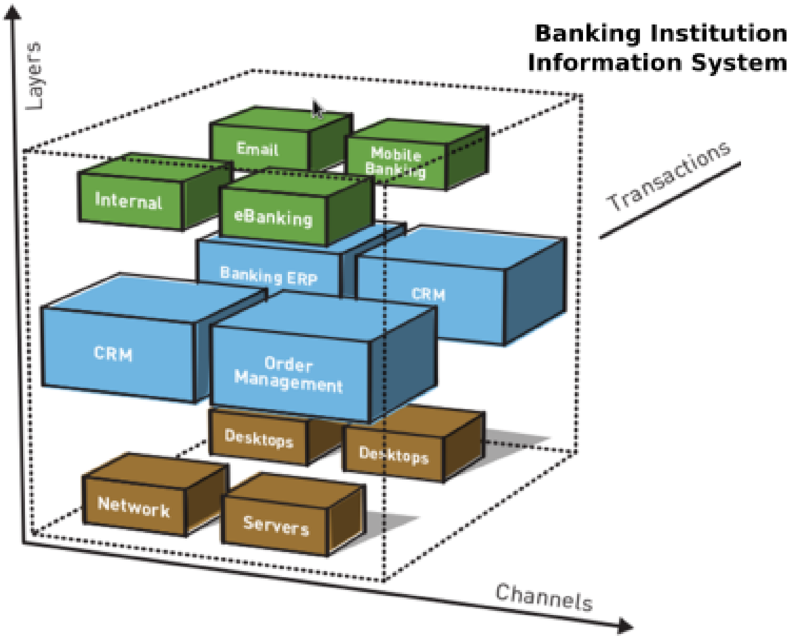
Summary

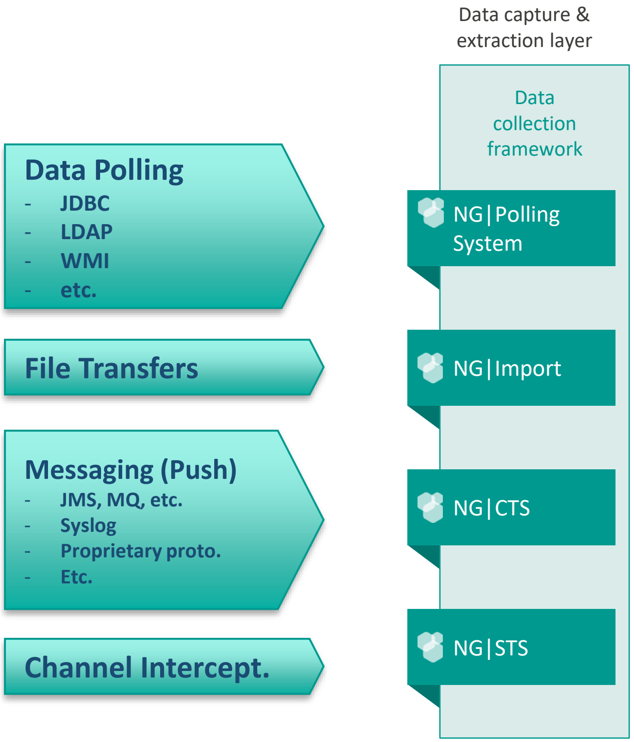
- Overview Data Collection Framework
- Flat File Import
- Database Polling

Application architecture



Data Collection Framework





Data capture & extraction layer

Data collection framework

Data Polling

- JDBC
- LDAP
- WMI
- etc.

File Transfers

Messaging (Push)

- JMS, MQ, etc.
- Syslog
- Proprietary proto.
- Etc.

Channel Intercept.

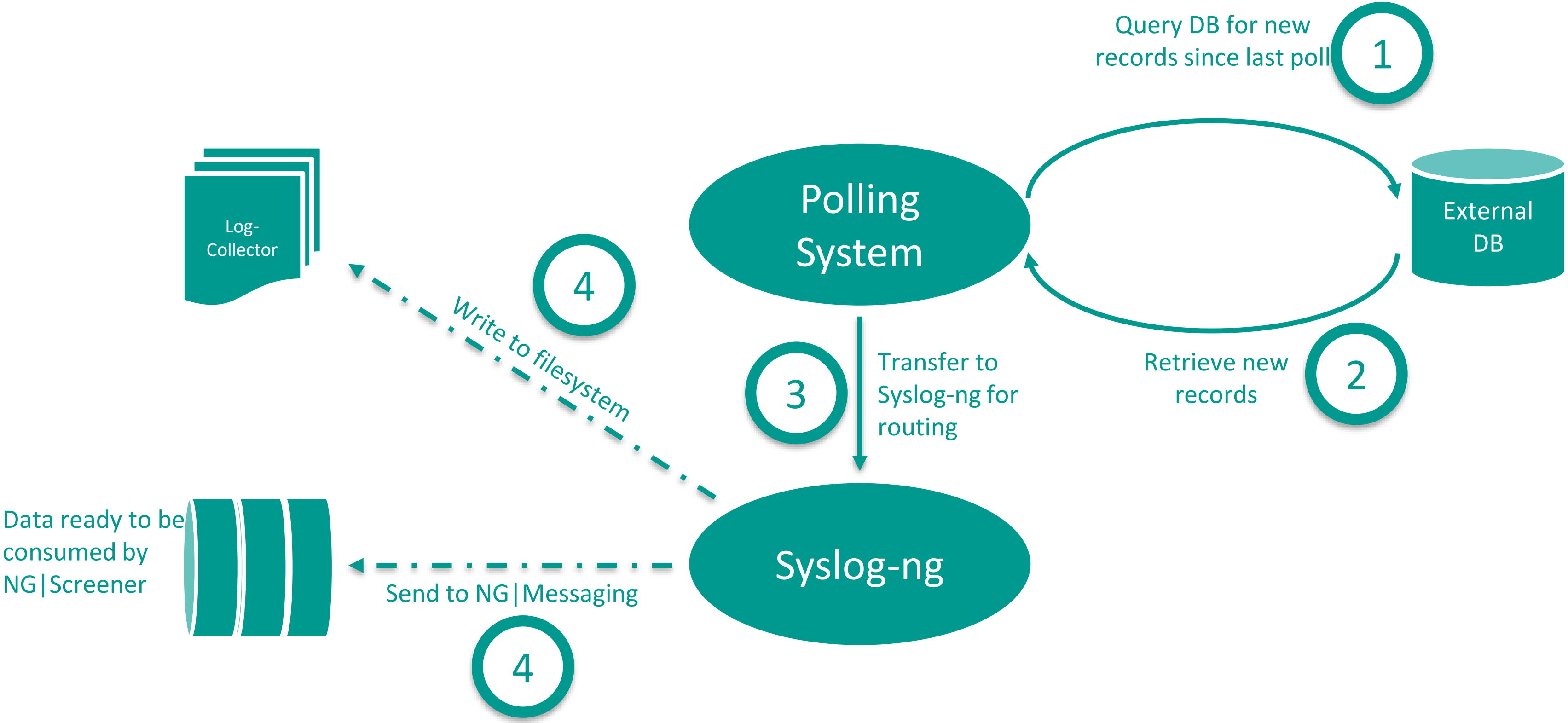
NG|Polling System

NG|Import

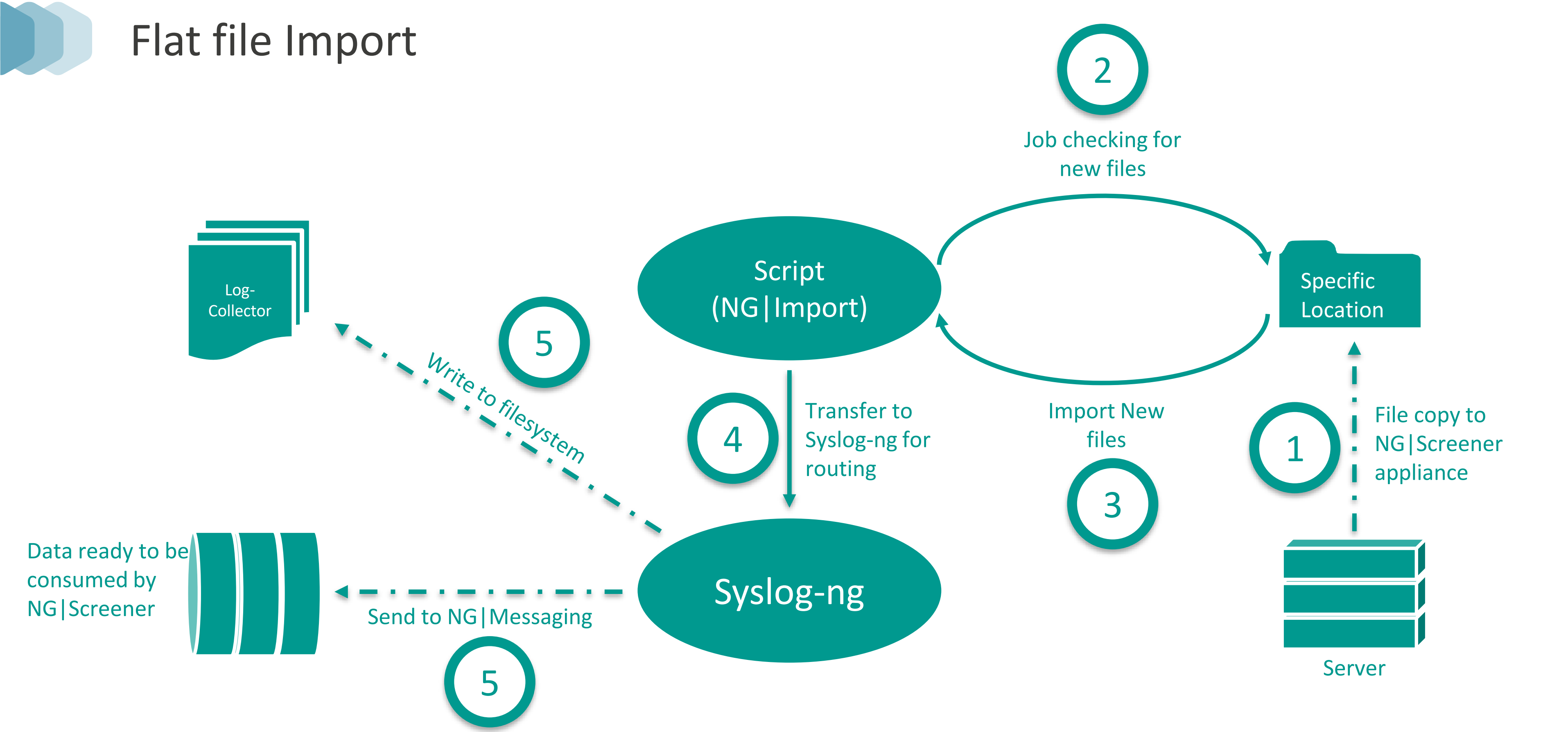
NG|CTS

NG|STS

Database Polling


Query DB for new records since last poll
1
Polling System
External DB
Retrieve new records
2
Log-Collector
4
Write to filesystem
3
Transfer to Syslog-ng for routing
Data ready to be consumed by NG|Screener
Syslog-ng
Send to NG|Messaging
4

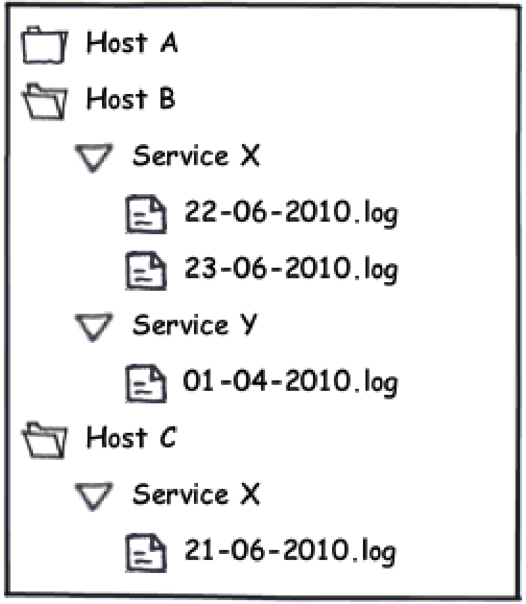
Flat file Import


2
Job checking for new files
Script (NG|Import)
Specific Location
Log-Collector
5
Write to filesystem
4
Transfer to Syslog-ng for routing
Import New files
3
1
File copy to NG|Screener appliance
Data ready to be consumed by NG|Screener
Syslog-ng
Send to NG|Messaging
5
Server

Data Collection Framework Overview

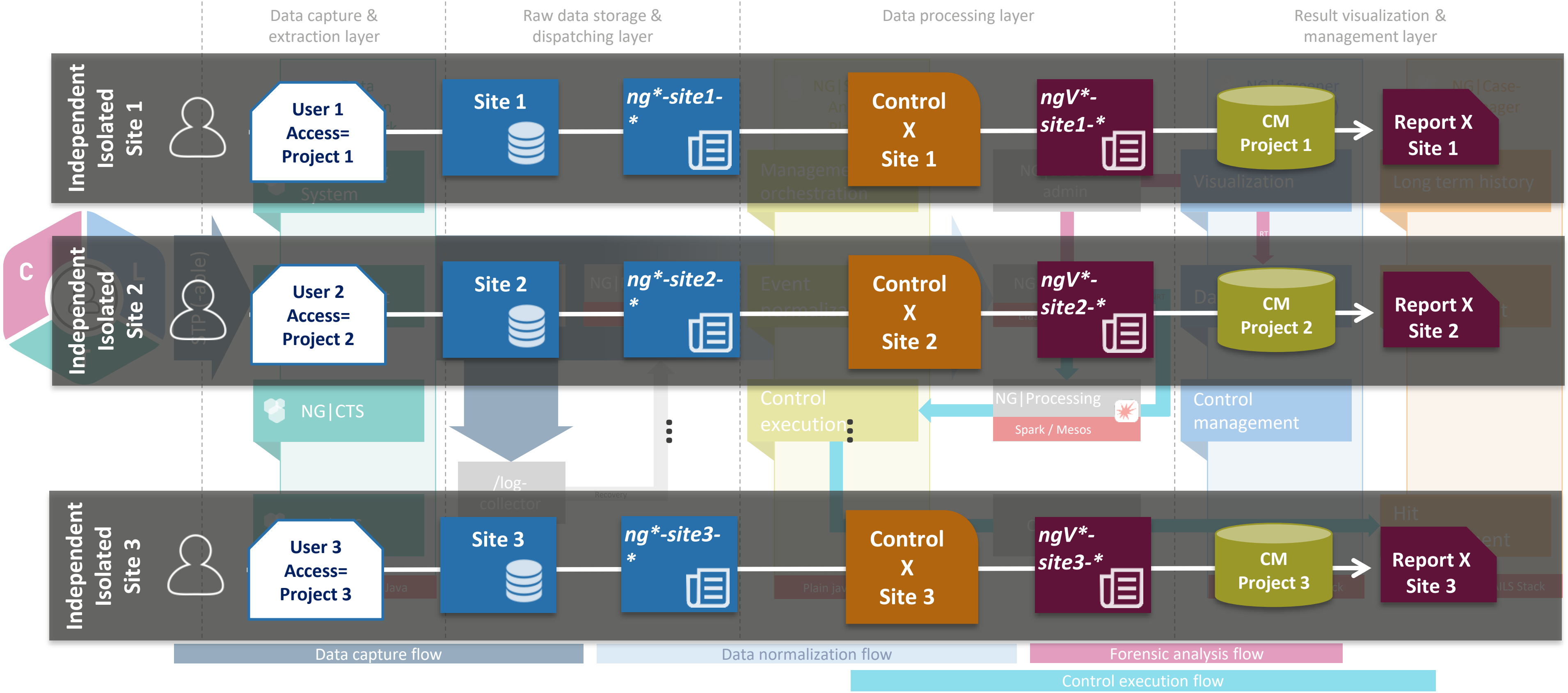
Data Storage

- Audit trails are centralized under /log-collector directory
    - Notion of Multi-tenancy in next slides
- This folder is structured by Year / Host / Service
- Filenames are formatted dd-mm-yyyy.log
- Files get compressed after 2 days to gain space
- Audit trails are compliant with Syslog log format

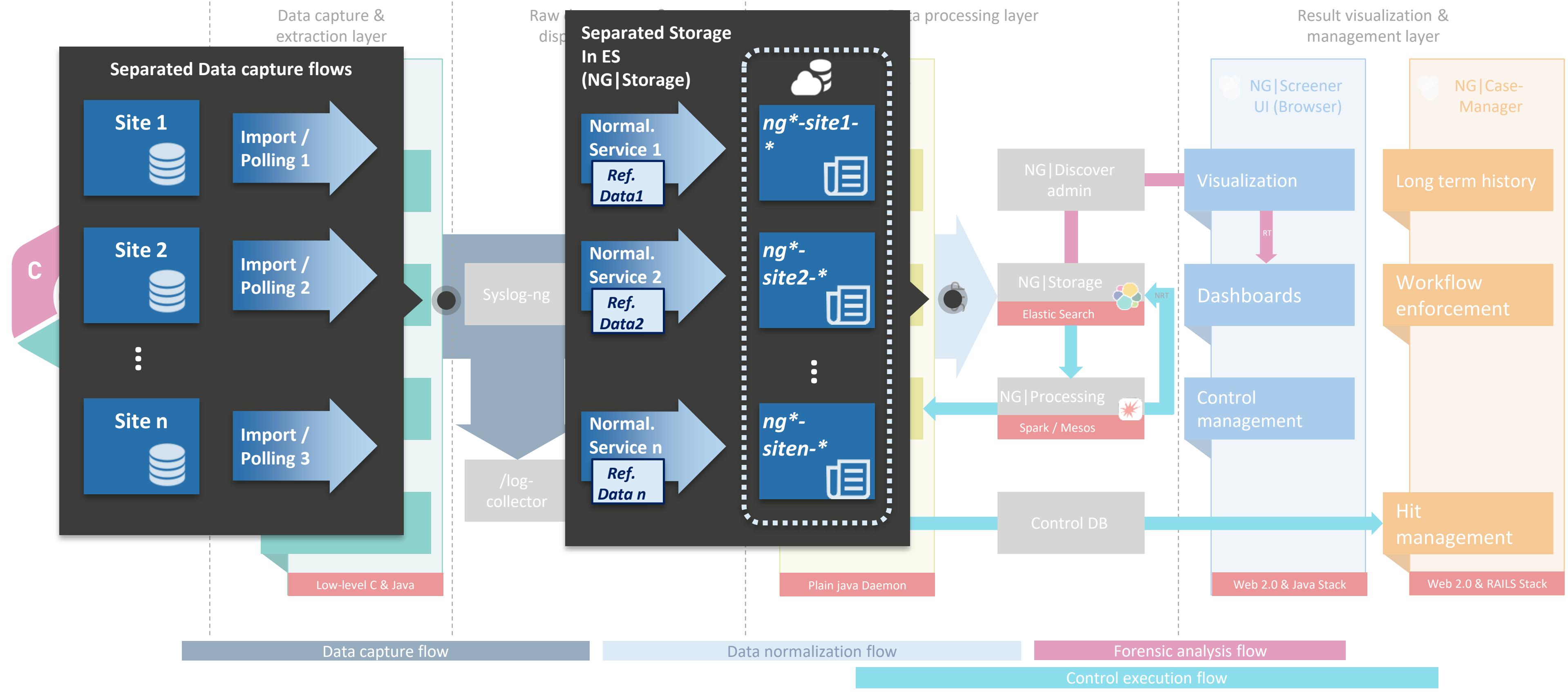


Raw Audit trails Storage

Multi-tenancy – Recap



Multi-tenancy – Data Ingestion segregation



Multi-tenancy – Data Ingestion segregation

- Key Principle
    - Different sites = different hosts in NG|Screener terminology
- Input Data
    - Different site data are imported by different independent mechanism
    - They are stored in specific host prefixed with tenant name
        - `/log-collector/year/host/service` with host being `tenantname_hostname`
- NG|Storage
    - Specific indexes created for each host
    - Secures access to only users or controls with proper access rights

RiskTech 100 2018

Gartner 2015 CoolVendor

Flat files
Data Capture

NG|Import

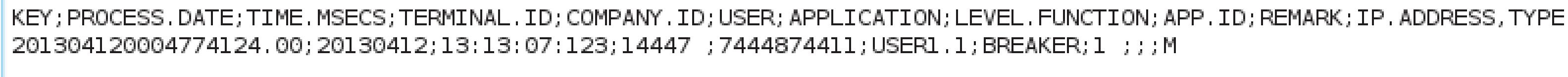
- Command line tool to Import several types of flat files to NG|Screener syslog file format

- Usage: `ngimport [options] [command] [command options]`
  - Options
    - -a Address of NG|Screener appliance
    - -c File path to be imported (with ‘file:’ before path)
    - -v Verbose mode, give some information in case of error
    - -h Show help
  - Commands:
    - t24Protocol Import T24 Protocol file
    - t24Journal Import T24 Journal file (Transaction or Overrides)
    - CSVFile Import CSV file

Temenos T24 Protocol Example

- Show help:
  - `ngimport -a localhost -c file:/tmp/test.txt t24Protocol -h`

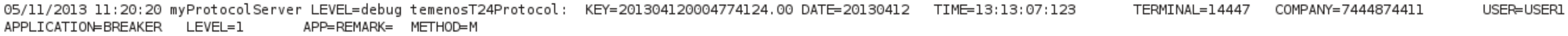
- Input file example:



```
KEY;PROCESS.DATE;TIME.MSECS;TERMINAL.ID;COMPANY.ID;USER;APPLICATION;LEVEL.FUNCTION;APP.ID;REMARK;IP.ADDRESS,TYPE
20130412000477412 4.00;20130412;13:13:07:123;14447 ;7444874411;USER1.1;BREAKER;1 ;;;M
```

- Import command:
  - `ngimport -c file:/home/admin/PROTOCOL.txt -a localhost t24Protocol -s DEFAULT_myProtocolServer`

- Output:
  - Located in: `/log-collector/2013/DEFAULT_myProtocolServer/temenosT24Protocol/23-11-2013.log`

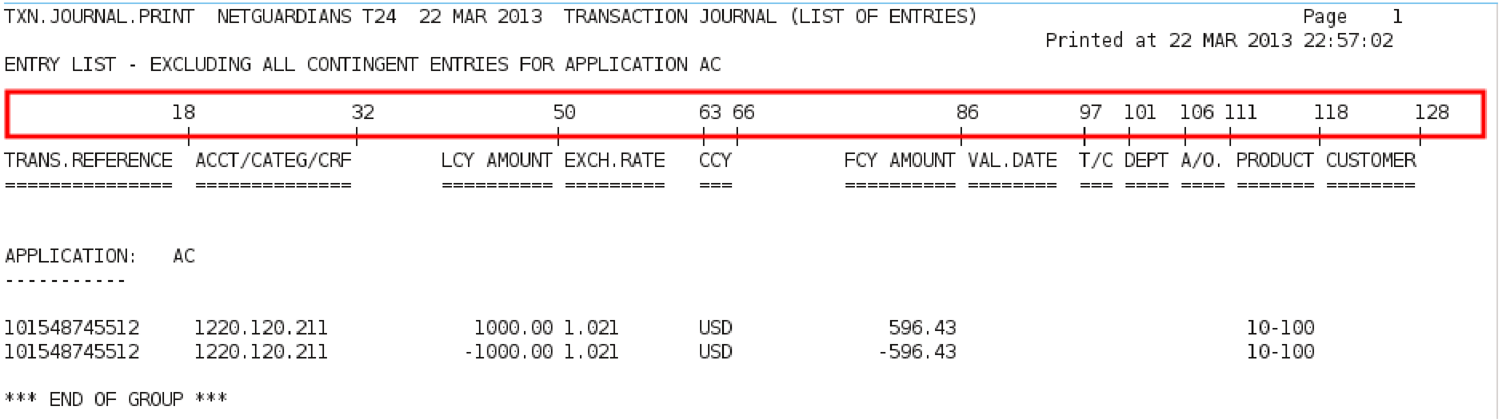


```
05/11/2013 11:20:20 myProtocolServer LEVEL=debug temenosT24Protocol:  KEY=20130412000477412 4.00 DATE=20130412   TIME=13:13:07:123       TERMINAL=14447   COMPANY=7444874411       USER=USER1
APPLICATION=BREAKER   LEVEL=1       APP=REMARK=  METHOD=M
```

Temenos T24 Transaction Example (1/2)

- Show help:
  - `ngimport -a localhost -c file:/tmp/test.txt t24Journal -h`

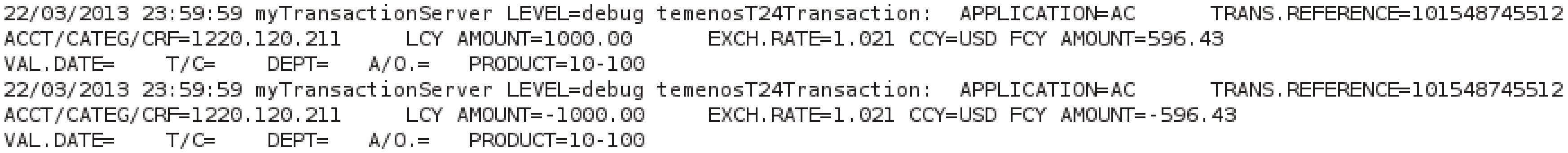
- Input file example (in red column width information, here example for T24 model bank):



```
TXN.JOURNAL.PRINT  NETGUARDIANS T24  22 MAR 2013  TRANSACTION JOURNAL (LIST OF ENTRIES)                         Page    1
                                                                                           Printed at 22 MAR 2013 22:57:02
ENTRY LIST - EXCLUDING ALL CONTINGENT ENTRIES FOR APPLICATION AC

               18             32                50           63 66                 86         97  101  106 111    118      128

TRANS.REFERENCE  ACCT/CATEG/CRF       LCY AMOUNT EXCH.RATE   CCY         FCY AMOUNT VAL.DATE  T/C DEPT A/O. PRODUCT CUSTOMER
===============  ==============       ========== =========   ===         ========== ========  === ==== ==== ======= ========


APPLICATION:   AC
-----------

101548745512     1220.120.211            1000.00 1.021       USD             596.43                          10-100
101548745512     1220.120.211           -1000.00 1.021       USD            -596.43                          10-100

*** END OF GROUP ***
```

Temenos T24 Transaction Example (2/2)

- Import command:
  - `ngimport -c file:/home/admin/TRANSACTION.txt -a localhost t24Journal -sn "temenosT24Transaction" -s DEFAULT_myTransactionServer --column-width "18 ,32 ,50 ,63 ,66 ,86 ,97 ,101 ,106 ,111 ,118 ,128“`
- Output:
  - Located in: `/log-collector/2013/DEFAULT_myTransactionServer/temenosT24Transaction/26-11-2013.log`

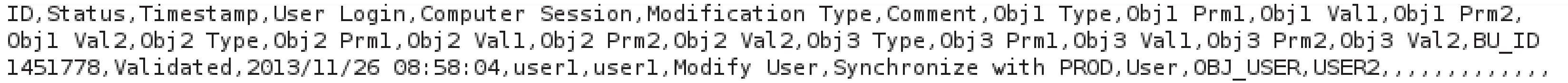


```
22/03/2013 23:59:59 myTransactionServer LEVEL=debug temenosT24Transaction:  APPLICATION=AC      TRANS.REFERENCE=101548745512
ACCT/CATEG/CRF=1220.120.211     LCY AMOUNT=1000.00      EXCH.RATE=1.021 CCY=USD FCY AMOUNT=596.43
VAL.DATE=    T/C=    DEPT=   A/O.=   PRODUCT=10-100
22/03/2013 23:59:59 myTransactionServer LEVEL=debug temenosT24Transaction:  APPLICATION=AC      TRANS.REFERENCE=101548745512
ACCT/CATEG/CRF=1220.120.211     LCY AMOUNT=-1000.00     EXCH.RATE=1.021 CCY=USD FCY AMOUNT=-596.43
VAL.DATE=    T/C=    DEPT=   A/O.=   PRODUCT=10-100
```

CSV File Example (1/3)

- Show help:
  - `ngimport -a localhost -c file:/tmp/test.txt CSVFile -h`

- Input file example:



```
ID,Status,Timestamp,User Login,Computer Session,Modification Type,Comment,Obj1 Type,Obj1 Prm1,Obj1 Val1,Obj1 Prm2,
Obj1 Val2,Obj2 Type,Obj2 Prm1,Obj2 Val1,Obj2 Prm2,Obj2 Val2,Obj3 Type,Obj3 Prm1,Obj3 Val1,Obj3 Prm2,Obj3 Val2,BU_ID
1451778,Validated,2013/11/26 08:58:04,user1,user1,Modify User,Synchronize with PROD,User,OBJ_USER,USER2,,,,,,,,,,,,,
```

- Import command:
  - `ngimport -c file:/home/ng-dev/CSVFILE.csv -a localhost CSVFile -cf /home/ng-dev/config.properties -s DEFAULT_myCSVServer -df ’yyyy/MM/dd hh:mm:ss’ -sepa ’,’`

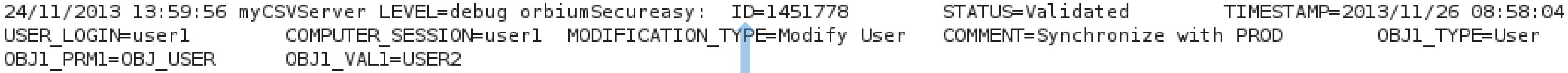
CSV File Example (2/3)

- Example CSV properties file

```
# Comment
COLUMN1 =ID
COLUMN2 = STATUS
DATE = TIMESTAMP
COLUMN3 = USER_LOGIN
COLUMN4 = COMPUTER_SESSION
COLUMN5 = MODIFICATION_TYPE
COLUMN6 = COMMENT
COLUMN7 = OBJ1_TYPE
…
COLUMN18 = OBJ3_PRM1
COLUMN19 = OBJ3_VAL1
COLUMN20 = OBJ3_PRM2
COLUMN21 = OBJ3_VAL2
COLUMN22 = BU_ID
CONCAT_DATE_AND_TIME =NO
SERVICE = orbiumSecureasy
```

CSV File Example (3/3)

- Output
  - Located in: `/log-collector/2013/DEFAULT_myCSVServer/orbiumSecureasy/24-11-2013.log`



```
24/11/2013 13:59:56 myCSVServer LEVEL=debug orbiumSecureasy:  ID=1451778        STATUS=Validated        TIMESTAMP=2013/11/26 08:58:04
USER_LOGIN=user1        COMPUTER_SESSION=user1  MODIFICATION_TYPE=Modify User   COMMENT=Synchronize with PROD        OBJ1_TYPE=User
OBJ1_PRM1=OBJ_USER      OBJ1_VAL1=USER2
```

```
# Comment
COLUMN1 = ID
COLUMN2 = STATUS
DATE = TIMESTAMP
COLUMN3 = USER_LOGIN
COLUMN4 = COMPUTER_SESSION
COLUMN5 = MODIFICATION_TYPE
COLUMN6 = COMMENT
COLUMN7 = OBJ1_TYPE
…
```

NetGuardians

RiskTech 100 2018

Gartner 2015 CoolVendor

Database Polling
Data Capture

swiss made software

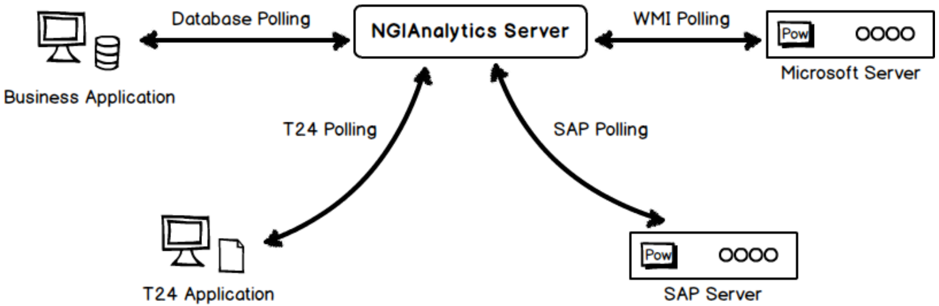
Polling Overview

7 types of polling

- JDBC
- T24
- FlexCube
- REST
- SAP
- WMI
- LDAP

3 Different Methods

- Fetch all
  - Table is truncated between two polls
- Fetch with status
  - Status field should be available
- Fetch and delete
  - Write access on table



Polling Configuration

- Polling configuration files location:
  - /etc/ng-screener/polling-system/targets/[jdbcTargets|ldapTargets|mswmiTargets|sapTargets|newT24Targets|flexcubeTargets|restTargets]

- Polling configurations examples are available in:
  - /etc/ng-screener/polling-system/targets (usual sample to be used)
  - /usr/local/ng-screener/connectors/connector-connectorName/polling (More specific examples)

- Polling status:
  - /etc/ng-screener/polling-system/status
    - service@host.pollstatus.json (stored status for fetch with status mode)
    - service@host.nextpoll.json (date/time of the next poll)
- ngadmin useful commands
  - `ngadmin --tenant=TENANT_NAME polling_listStatus`

- Polling logs:
  - /var/log/ng-screener/polling-system/polling-system.log

Polling configuration – Command line

- Connect with SSH client to NGScreener server
  - Connect as admin user first
  - Then escalate to root

- Go to polling configuration directory
  - `cd /etc/ng-screener/polling-system/targets/`

- Copy a sample to the correct subdirectory
  - Example T24
    - `cp newT24_sample.conf newT24Targets/myT24polling.conf`

- Adapt the file to your setup
  - `vim newT24Targets/myT24polling.conf`

- Restart polling-system and check polling logs for errors
  - `systemctl restart polling-system`
  - `tail -f /var/log/ng-screener/polling-system/polling-system.log`

Polling Configuration

- Special Cases
  - New T24 Targets
    - Duplicate Detection
  - History table polling

- New T24 Targets
  - Eliminate duplicates that may occur
  - Robust to missing entries

- History table polling
  - Data source provides information about change
    - But sometimes not the change itself
  - Detect and retrieve changes made to objects in DB

New T24 Targets

Designed to

- Eliminate duplicates that may occur
- Be Robust to missing entries
- Similar functionality available for JDBC polling as well

Work by

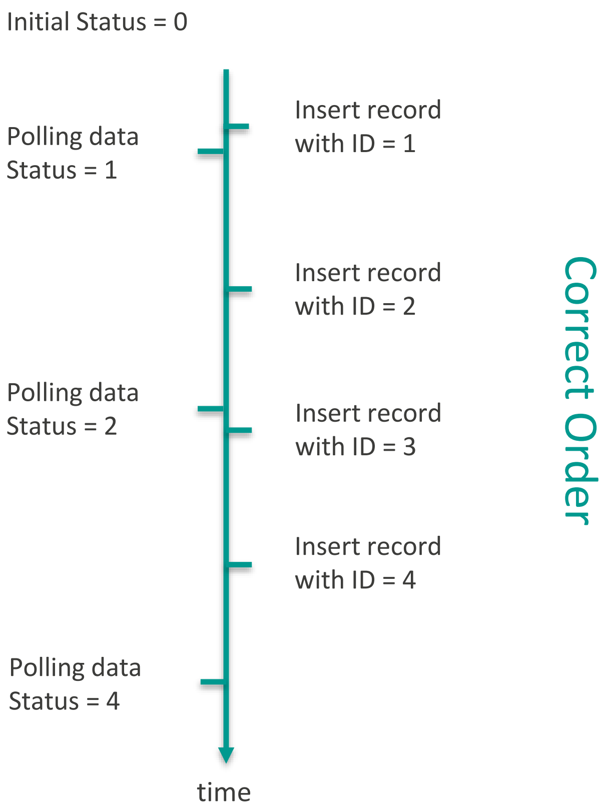
- Caching collected RECIDs
- First stage Query to get list of new RECIDs
- Second stage Query to get data for RECIDs not already collected

Duplicate Detection

- Sometimes entries are not inserted in table in the correct order

- Using only status will make system miss some data

- Need of a cache of collected data and overlap of polling

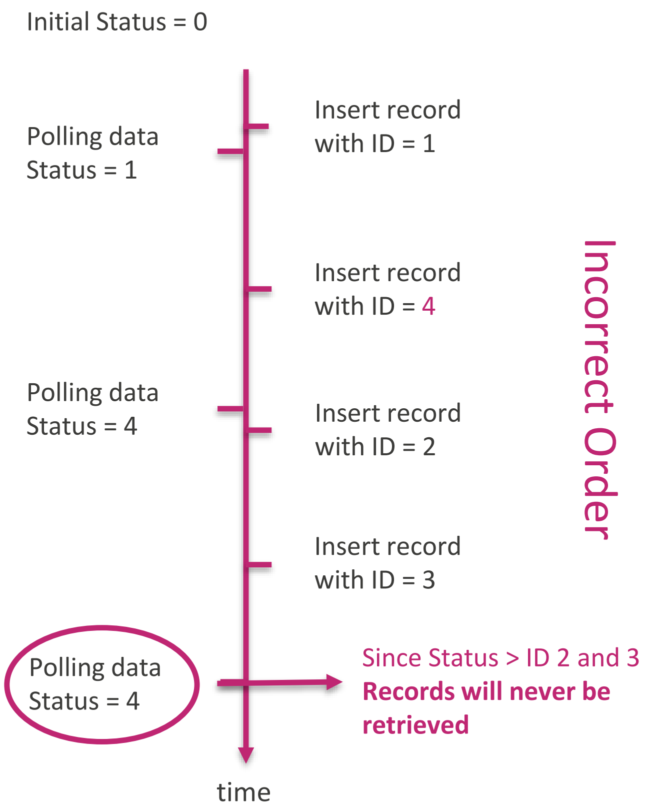


Duplicate Detection

- Sometimes entries are not inserted in table in the correct order

- Using only status will make system miss some data

- Need of a cache of collected data and overlap of polling



Initial Status = 0

Polling data
Status = 1

Insert record
with ID = 1

Insert record
with ID = 4

Polling data
Status = 4

Insert record
with ID = 2

Insert record
with ID = 3

Incorrect Order

Polling data
Status = 4

Since Status > ID 2 and 3
Records will never be retrieved

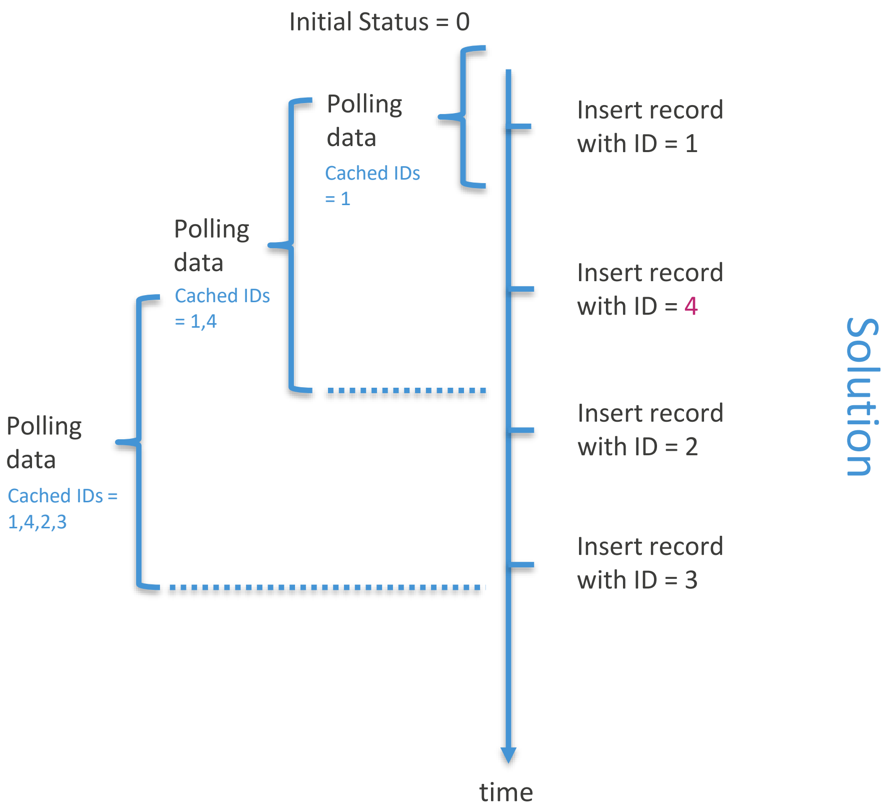
time

Duplicate Detection

- Sometimes entries are not inserted in table in the correct order

- Using only status will make system miss some data

- Need of a cache of collected data and overlap of polling



New T24 targets configuration

- Specific parameters
  - **QueryForDataTemplate**: 2nd Stage Query to retrive actual data from a batch of RECIDs (where RECID IN …)
  - **TimeInSecondsOffsetLogFetching**: Time to consider looking backwards for new rows
  - **TimeInSecondsCacheConservation**: Keys conservation time in cache. Should be always greater than TimeInSecondsOffsetLogFetching
  - **TimeInSecondsInitOffset**: Time offset for the first poll (when no status is stored)
  - **dataFetchBatchSize**: Number of rows to fetch in a single batch (single access to DB)
- Example and explanation is provided in sample configuration in polling-system directory

History table polling

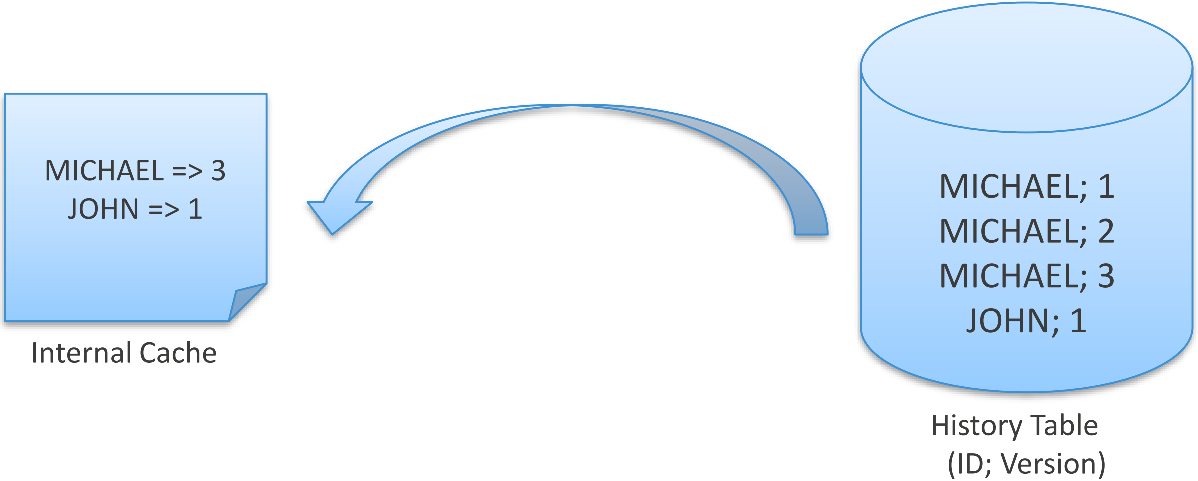
Designed to

- Detect and retrieve changes made to objects in DB
- **Example**: Changes made on Users in T24

Work by

- Defining a postprocess to usual polling to compare values in “Current” and “History” tables
  - Current and History tables are related to T24 terminology
  - Information could be stored on same table

History table polling - Initialization



MICHAEL => 3
JOHN => 1

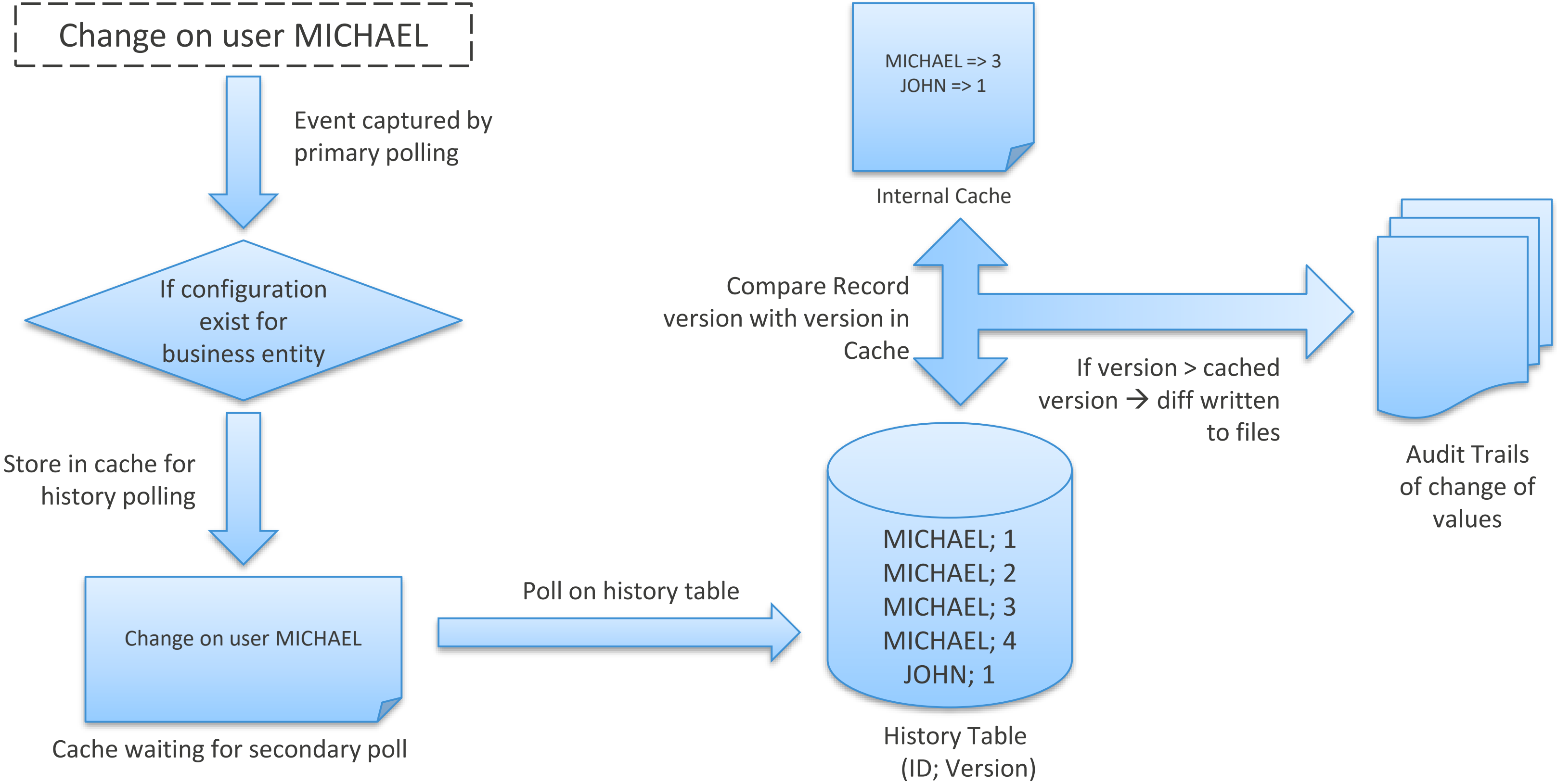
Internal Cache

MICHAEL; 1
MICHAEL; 2
MICHAEL; 3
JOHN; 1

History Table
(ID; Version)

Retrieve maximum version from History table and store it in internal cache

History table polling – Run time



Change on user MICHAEL

Event captured by primary polling

If configuration exist for business entity

Store in cache for history polling

Change on user MICHAEL

Cache waiting for secondary poll

Poll on history table

MICHAEL; 1
MICHAEL; 2
MICHAEL; 3
MICHAEL; 4
JOHN; 1

History Table (ID; Version)

MICHAEL => 3
JOHN => 1

Internal Cache

Compare Record version with version in Cache

If version > cached version → diff written to files

Audit Trails of change of values

History table polling

- Configuration
  - Check Chapter 6 of Polling System Administration Guide

THANK YOU!

Contact us

☎ +41 24 425 97 60

✉ info@netguardians.ch

www.netguardians.ch

Linkedin.com/company/netguardians

Facebook.com/NetGuardians

@netguardians

https://www.youtube.com/netguardians

NetGuardians Headquarters

Y-Parc, Av. des Sciences 13
1400 Yverdon-les-Bains
Switzerland

T +41 24 425 97 60
F +41 24 425 97 65

NetGuardians Africa

KMA Centre , 7th floor,
Mara Road Upper Hill,
Nairobi, Kenya

T +254 204 93 11 96

NetGuardians Asia

143 Cecil Street
#09-01 GB Building
069542 Singapore

T +65 6224 0987

NetGuardians
Eastern Europe

Koszykowa 61, 00-667
Warsaw, Poland

NetGuardians Germany

Rhein-Main Gebiet
Germany

T +49 172 3799003