

# Feature Engineering to produce Process Mining Data for Clinical Order Processes from Electronic Health Record (EHR) Systems

Hilda B. Klasky

Oak Ridge National Laboratory

Process Mining Camp, June 2020

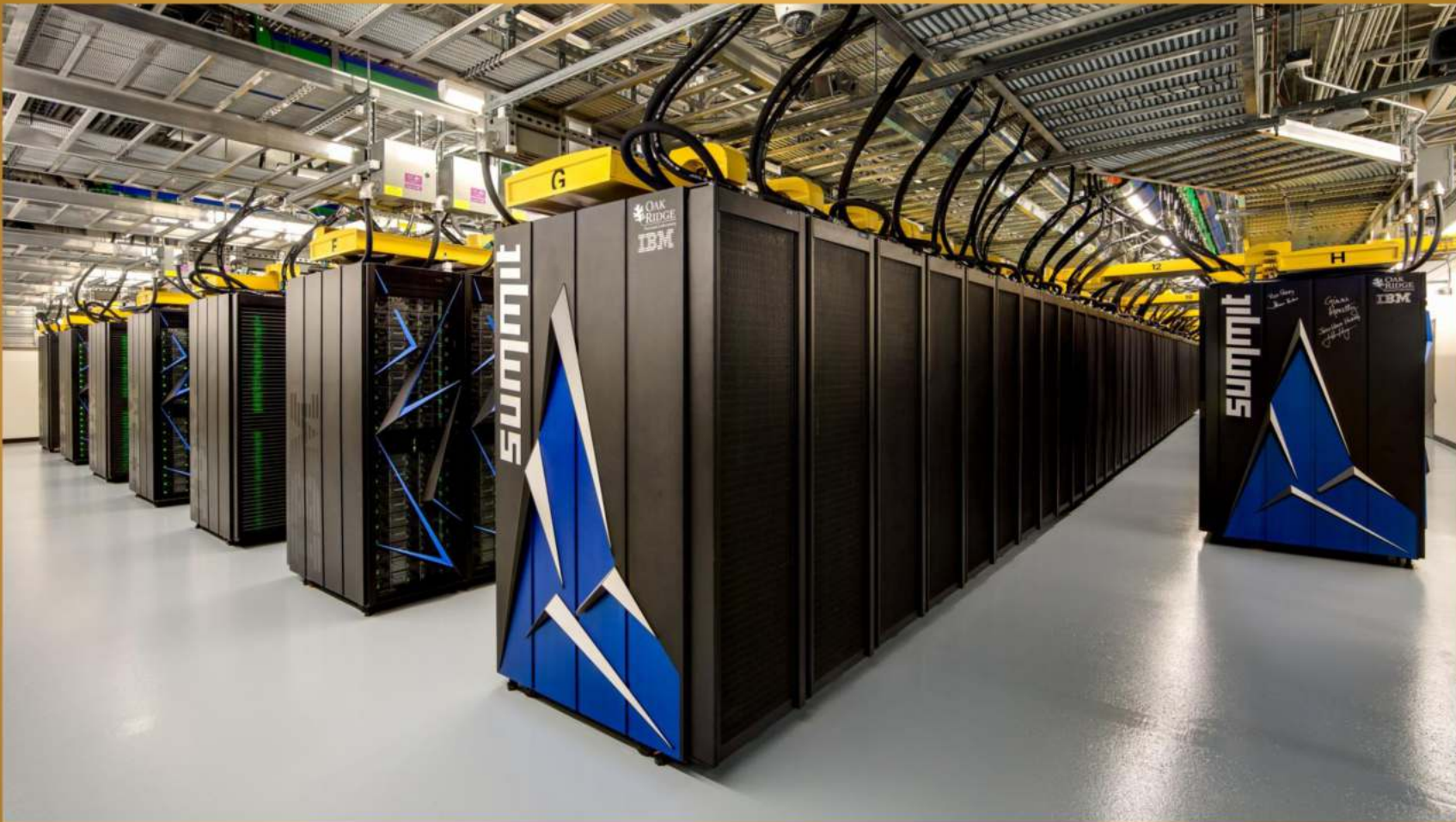
ORNL is managed by UT-Battelle, LLC for the US Department of Energy













- **VA Mission Statement:**

*To fulfill President Lincoln's promise "To care for him who shall have borne the battle, and for his widow, and his orphan" by serving and honoring the men and women who are America's veterans.*

- VA carries out **four specific missions** to make good on that commitment.

1. **Veterans Health Care**
2. **Veterans Benefits**
3. **National Cemeteries**
4. **Improve the Nation's preparedness**



- **VA Vision:**

*To provide veterans the world-class benefits and services they have earned - and to do so by adhering to the highest standards of compassion, commitment, excellence, professionalism, integrity, accountability, and stewardship.*







**VA**



U.S. Department  
of Veterans Affairs



## 'I knew something was not right': Mass cancellations of diagnostic test orders at VA hospitals draw scrutiny

Donovan Slack, USA TODAY

Published 6:00 a.m. ET Oct. 1, 2018 | Updated 4:28 p.m. ET Oct. 1, 2018



Veterans Health  
Administration

# Patient Safety Alert

Issued by VHA Central Office

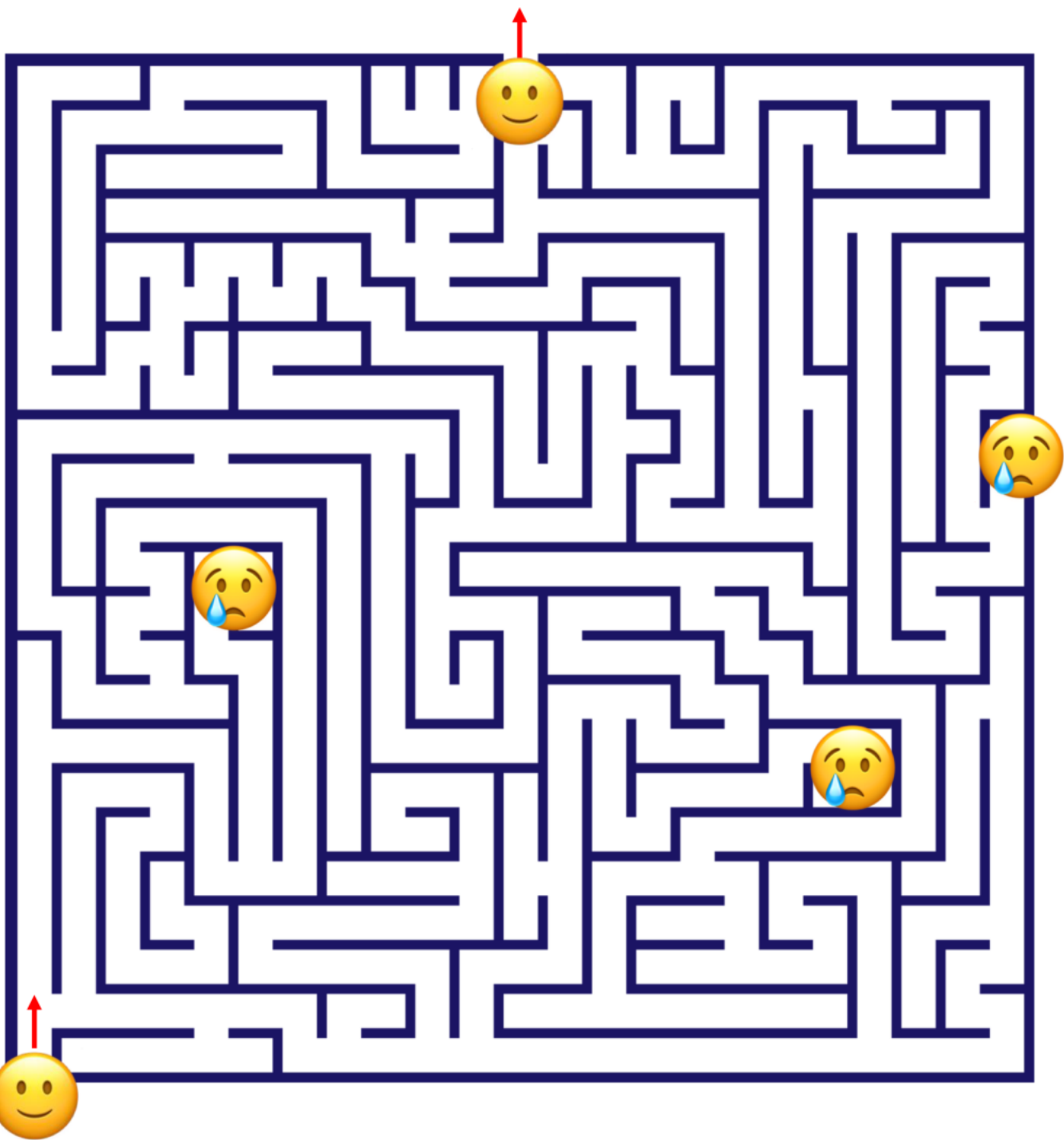
AL18-04

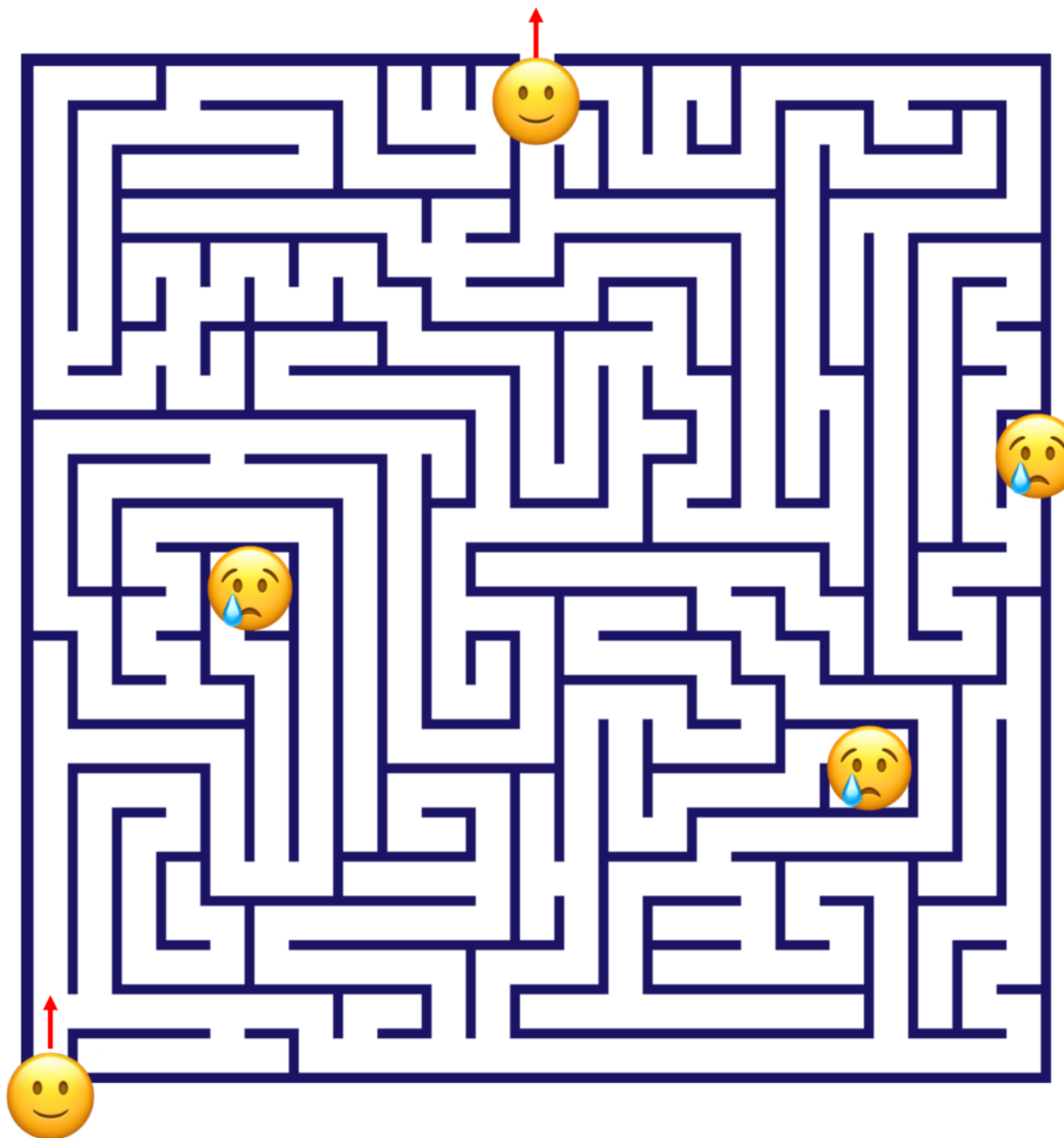
Item:

CPRS Imaging orders may have been rejected and potentially overlooked

June 14, 2018







**EHR Systems are complex as a maze**

**Most clinical orders can find the successful path to the end**

**But, some may get stuck on a dead-end.**

**This study aims to identify successful orders and not successful ones.**

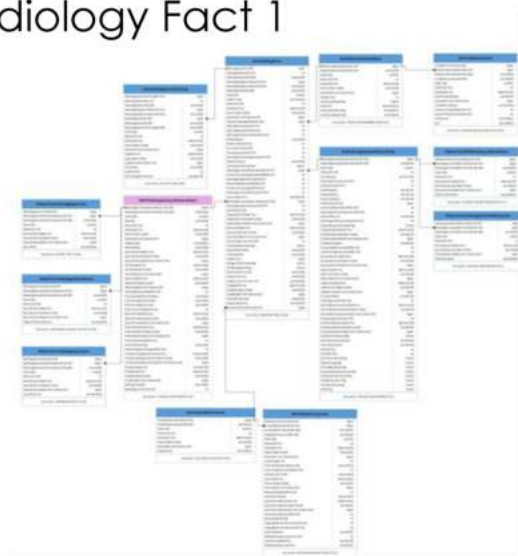


The **hazard** that we are trying to identify in this study is:



**the intended treatment of the patient is not fully completed.**

Radiology Fact 1



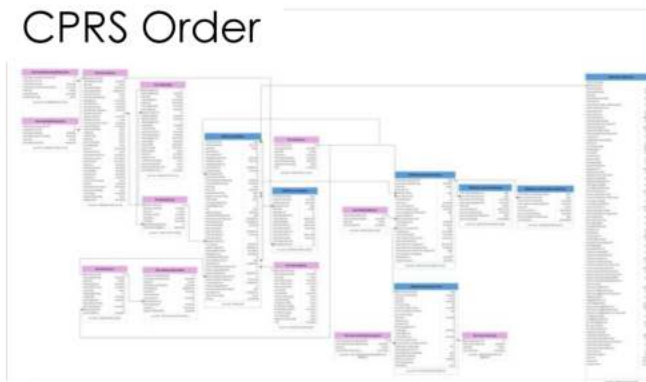
Lab Microbiology



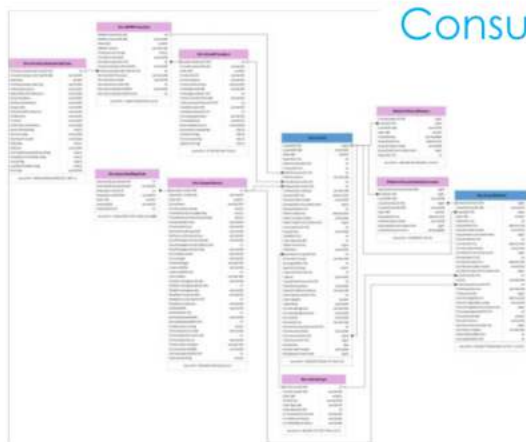
Pharmacy Outpatient 2.2



CPRS Order



Consult 2.1



LabChem Fact



Radiology 1.0 Fact 2

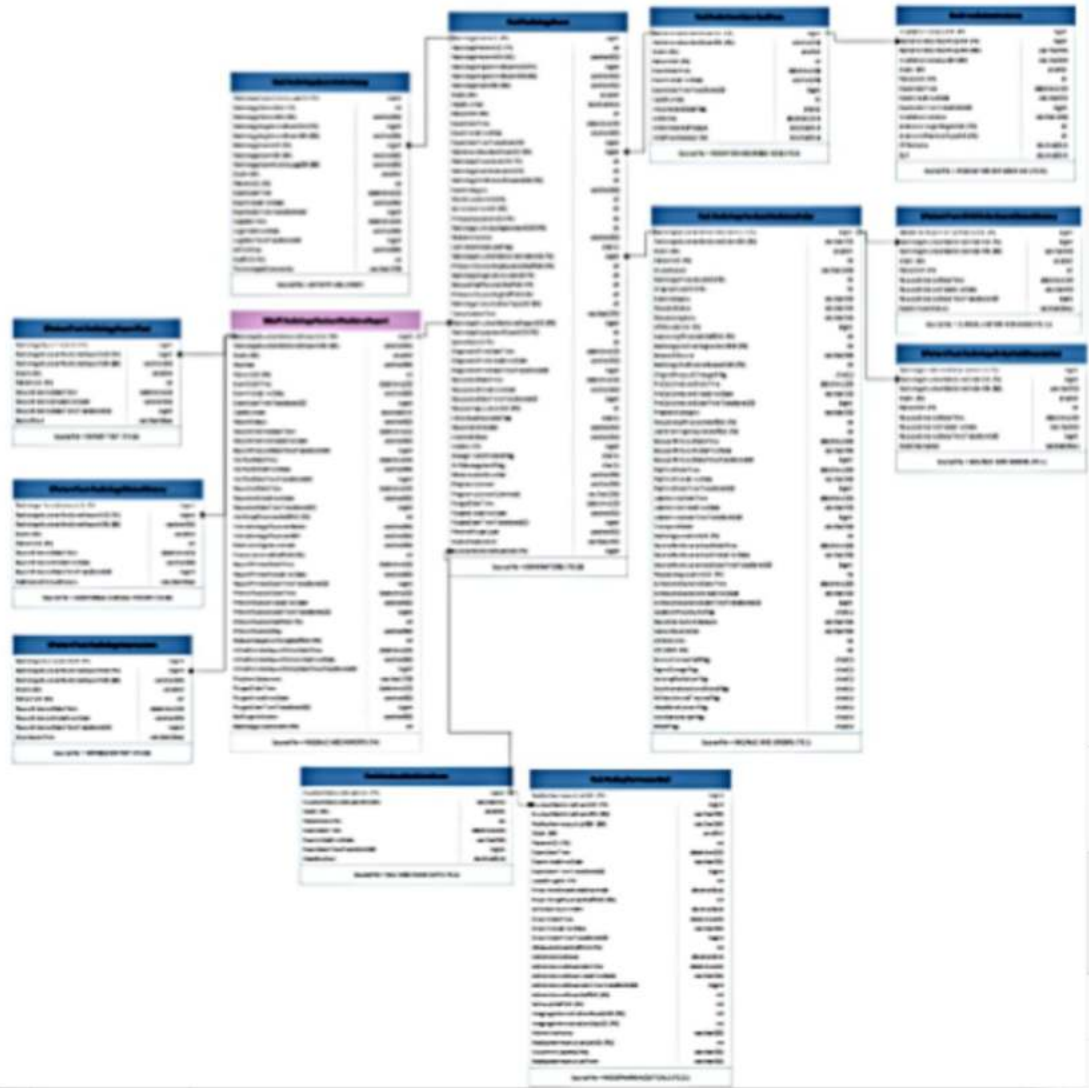




Radiology Fact 1



Radiology Fact 1



Pharmacy Outpatient 2.2



We identified and included in our study ALL date columns, statuses, actions and activities for the orders in this data domain

EnteredDateTime	smalldatetime
EnteredVistaDate	varchar(50)
OrderStatusSID (FK)	int

E  
X  
A  
M  
P  
L  
E

- **Date column:**
  - EnteredDateTime, 2017-04-06 15:37:00
- **Activity:**
  - EXAM ENTRY, 2017-04-06 15:42:00
- **Status:**
  - DISCONTINUED, COMPLETE, HOLD, PENDING, ACTIVE, SCHEDULED, UNRELEASED



# Horizontal sequence example: a long row of data

```
8314090 |DesiredNotGuaranteedDateTime,2017-04-06 00:00:00|  
ReportedDateTime,2017-04-06 00:00:00|RequestedDateTime,2017-04-06 00:00:00|  
OrderStartDateTime,2017-04-06 00:00:00|SignedDateTime,2017-04-06 15:37:00|  
EnteredDateTime,2017-04-06 15:37:00|OrderActionDateTime,2017-04-06 15:37:00|  
ReleaseDateTime,2017-04-06 15:37:00|RequestEnteredDateTime,2017-04-06 15:37:52|  
WAITING FOR EXAM,2017-04-06 15:42:00|StatusChangeDateTime,2017-04-06 15:42:00|  
ExamDateTime,2017-04-06 15:42:00|EXAM ENTRY,2017-04-06 15:42:00|  
StatusChangeDateTime,2017-04-06 15:47:00|EXAMINED,2017-04-06 15:47:00|  
EDIT BY CASE NO.,2017-04-06 15:47:00|VerifiedDateTime,2017-04-06 16:03:00|  
ReportEnteredDateTime,2017-04-06 16:03:00|  
LastActivityDateTime,2017-04-06 16:03:00|OrderStopDateTime,2017-04-06 16:03:00|  
COMPLETE,2017-04-06 16:03:00|StatusChangeDateTime,2017-04-06 16:03:00|  
ResultsDateTime,2017-04-06 16:03:00|VistaCreateDate,2017-04-06 17:55:10|  
VistaEditDate,2017-04-06 17:55:10
```

# Horizontal sequence example: a long row of data

```
8314090 |DesiredNotGuaranteedDateTime,2017-04-06 00:00:00|
ReportedDateTime,2017-04-06 00:00:00|RequestedDateTime,2017-04-06 00:00:00|
OrderStartDateTime,2017-04-06 00:00:00|SignedDateTime,2017-04-06 15:37:00|
EnteredDateTime,2017-04-06 15:37:00|OrderActionDateTime,2017-04-06 15:37:00|
ReleaseDateTime,2017-04-06 15:37:00|RequestEnteredDateTime,2017-04-06 15:37:52|
WAITING FOR EXAM,2017-04-06 15:42:00|StatusChangeDateTime,2017-04-06 15:42:00|
ExamDateTime,2017-04-06 15:42:00|EXAM ENTRY,2017-04-06 15:42:00|
StatusChangeDateTime,2017-04-06 15:47:00|EXAMINED,2017-04-06 15:47:00|
EDIT BY CASE NO.,2017-04-06 15:47:00|VerifiedDateTime,2017-04-06 16:03:00|
ReportEnteredDateTime,2017-04-06 16:03:00|
LastActivityDateTime,2017-04-06 16:03:00|OrderStopDateTime,2017-04-06 16:03:00|
COMPLETE,2017-04-06 16:03:00|StatusChangeDateTime,2017-04-06 16:03:00|
ResultsDateTime,2017-04-06 16:03:00|VistaCreateDate,2017-04-06 17:55:10|
VistaEditDate,2017-04-06 17:55:10
```

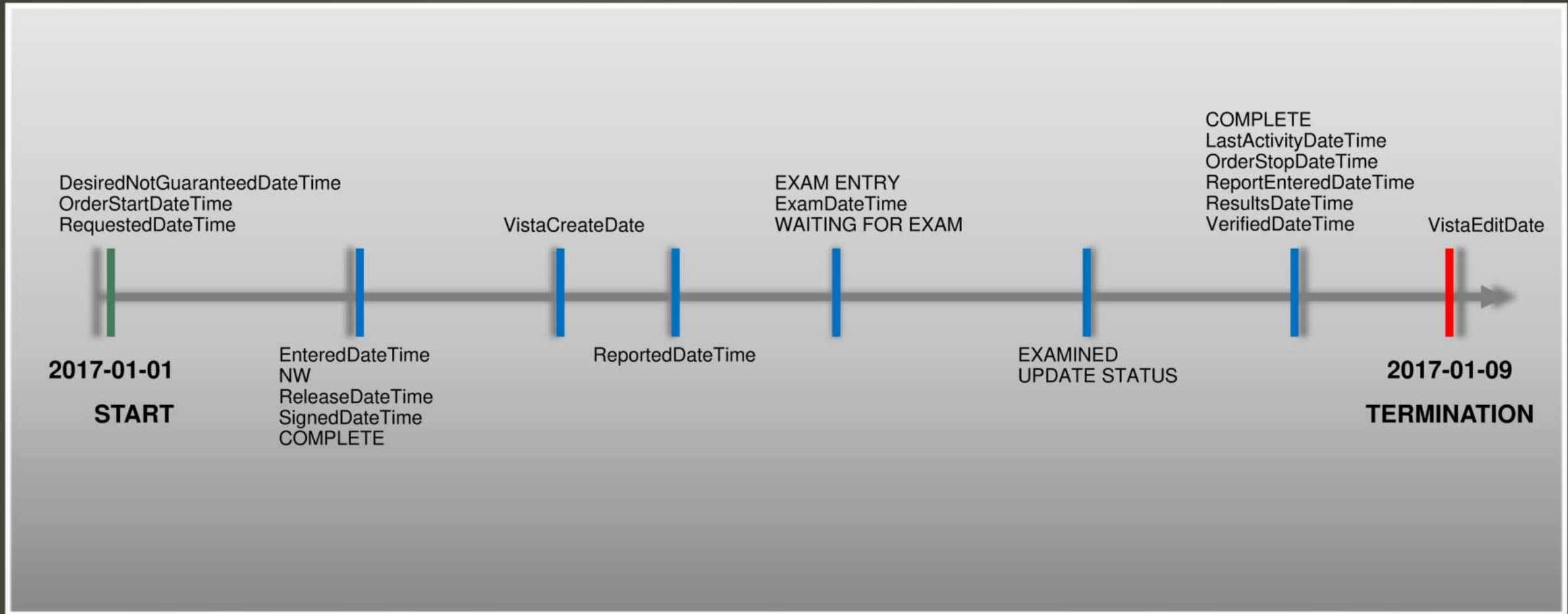
# Horizontal sequence example: a long row of data

**Deidentified case identification number**



```
8314090 | DesiredNotGuaranteedDateTime,2017-04-06 00:00:00 |  
ReportedDateTime,2017-04-06 00:00:00 | RequestedDateTime,2017-04-06 00:00:00 |  
OrderStartDateTime,2017-04-06 00:00:00 | SignedDateTime,2017-04-06 15:37:00 |  
EnteredDateTime,2017-04-06 15:37:00 | OrderActionDateTime,2017-04-06 15:37:00 |  
ReleaseDateTime,2017-04-06 15:37:00 | RequestEnteredDateTime,2017-04-06 15:37:52 |  
WAITING FOR EXAM,2017-04-06 15:42:00 | StatusChangeDateTime,2017-04-06 15:42:00 |  
ExamDateTime,2017-04-06 15:42:00 | EXAM ENTRY,2017-04-06 15:42:00 |  
StatusChangeDateTime,2017-04-06 15:47:00 | EXAMINED,2017-04-06 15:47:00 |  
EDIT BY CASE NO.,2017-04-06 15:47:00 | VerifiedDateTime,2017-04-06 16:03:00 |  
ReportEnteredDateTime,2017-04-06 16:03:00 |  
LastActivityDateTime,2017-04-06 16:03:00 | OrderStopDateTime,2017-04-06 16:03:00 |  
COMPLETE,2017-04-06 16:03:00 | StatusChangeDateTime,2017-04-06 16:03:00 |  
ResultsDateTime,2017-04-06 16:03:00 | VistaCreateDate,2017-04-06 17:55:10 |  
VistaEditDate,2017-04-06 17:55:10
```



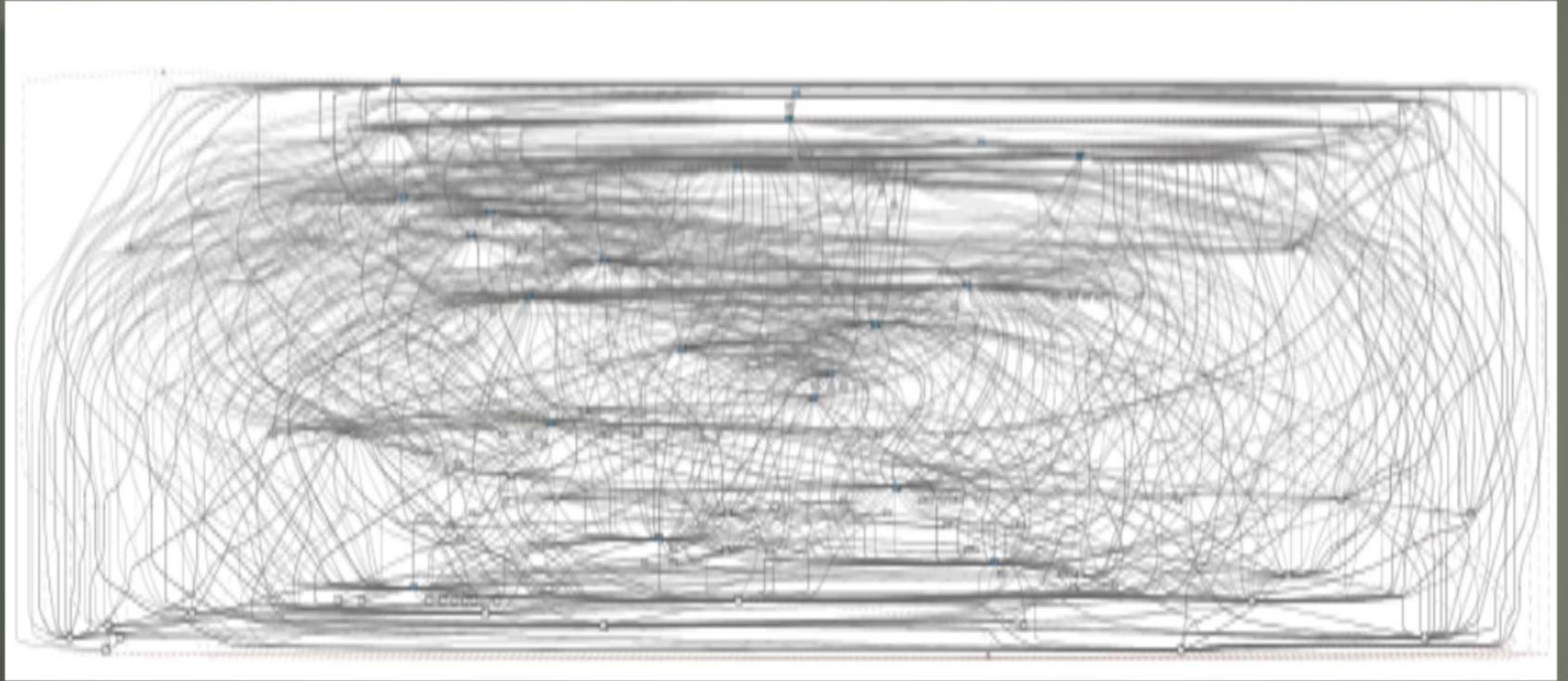


**Figure 1. Temporal clustering of a sample order illustrated on a timeline.**

# Vertical sequence example:

```
8314090,DesiredNotGuaranteedDateTime,2017-04-06 00:00:00
8314090,OrderStartDate,2017-04-06 00:00:00
8314090,ReportedDateTime,2017-04-06 00:00:00
8314090,RequestedDateTime,2017-04-06 00:00:00
8314090,EnteredDateTime,2017-04-06 15:37:00
8314090,NW,2017-04-06 15:37:00
8314090,ReleaseDateTime,2017-04-06 15:37:00
8314090,SignedDateTime,2017-04-06 15:37:00
8314090,COMPLETE,2017-04-06 15:37:52
8314090,EXAM ENTRY,2017-04-06 15:42:00
8314090,ExamDateTime,2017-04-06 15:42:00
8314090,WAITING FOR EXAM,2017-04-06 15:42:00
8314090,EDIT BY CASE NO.,2017-04-06 15:47:00
8314090,EXAMINED,2017-04-06 15:47:00
8314090,COMPLETE,2017-04-06 16:03:00
8314090,LastActivityDateTime,2017-04-06 16:03:00
8314090,OrderStopDateTime,2017-04-06 16:03:00
8314090,ReportEnteredDateTime,2017-04-06 16:03:00
8314090,ResultsDateTime,2017-04-06 16:03:00
8314090,VerifiedDateTime,2017-04-06 16:03:00
8314090,VistaCreateDate,2017-04-06 17:55:10
8314090,VistaEditDate,2017-04-06 17:55:10
```

Finding: the process map is complex...

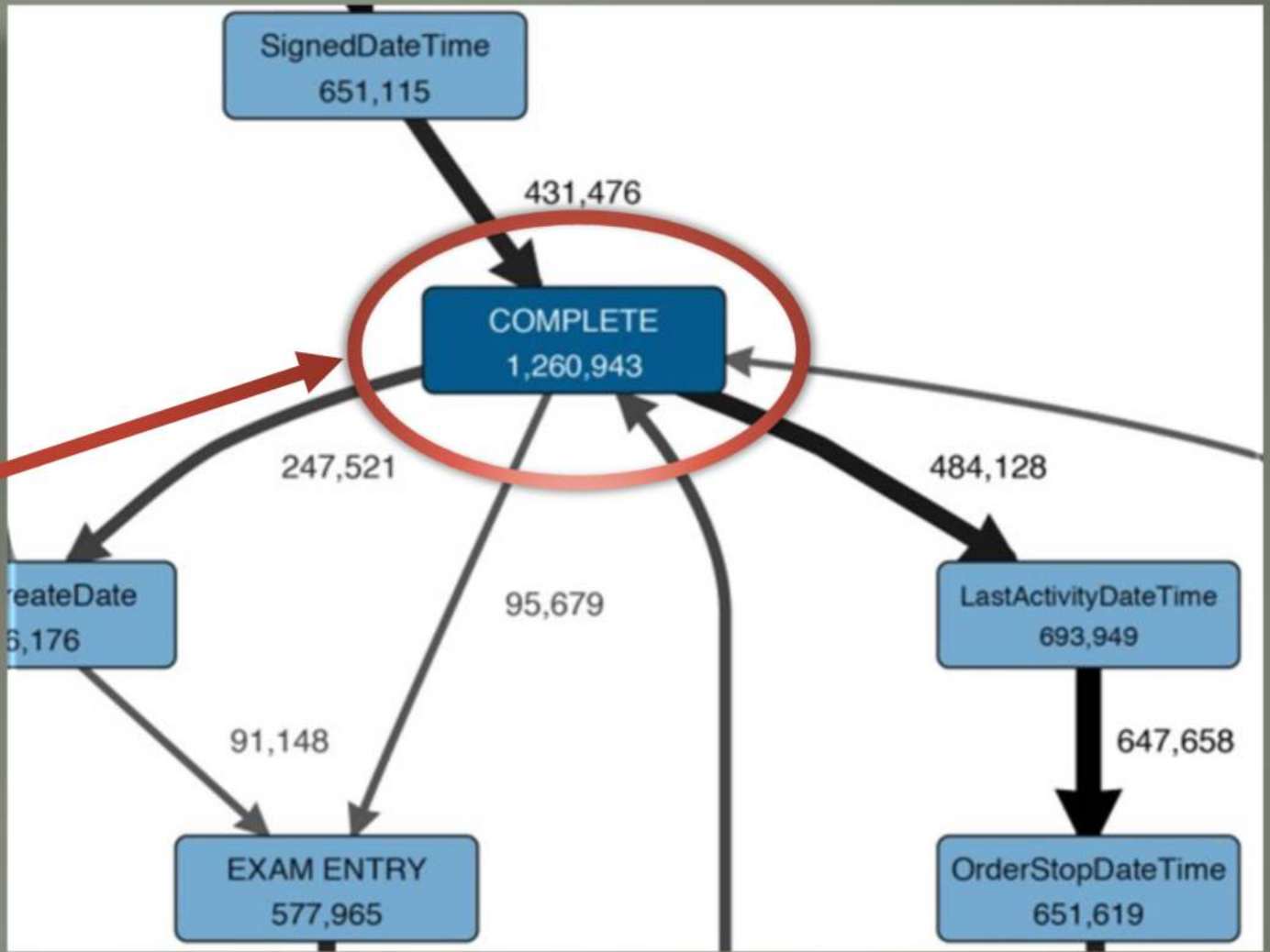
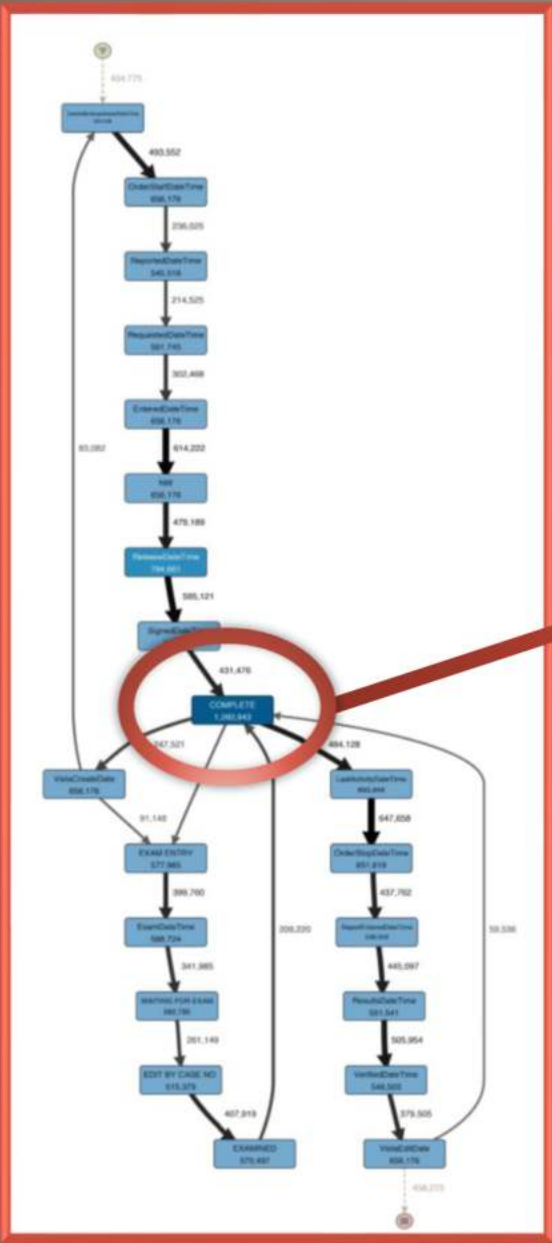






# Multiple COMPLETE events under different contexts

8314090,DesiredNotGuaranteedDateTime,2017-04-06 00:00:00  
8314090,OrderStartDate,2017-04-06 00:00:00  
8314090,ReportedDateTime,2017-04-06 00:00:00  
8314090,RequestedDateTime,2017-04-06 00:00:00  
8314090,EnteredDateTime,2017-04-06 15:37:00  
8314090,NW,2017-04-06 15:37:00  
8314090,ReleaseDateTime,2017-04-06 15:37:00  
8314090,signedDateTime,2017-04-06 15:37:00  
8314090,**COMPLETE**,2017-04-06 15:37:52  
8314090,EXAM ENTRY,2017-04-06 15:42:00  
8314090,ExamDateTime,2017-04-06 15:42:00  
8314090,WAITING FOR EXAM,2017-04-06 15:42:00  
8314090,EDIT BY CASE NO.,2017-04-06 15:47:00  
8314090,EXAMINED,2017-04-06 15:47:00  
8314090,**COMPLETE**,2017-04-06 16:03:00  
8314090,LastActivityDateTime,2017-04-06 16:03:00  
8314090,OrderStopDateTime,2017-04-06 16:03:00  
8314090,ReportEnteredDateTime,2017-04-06 16:03:00  
8314090,ResultsDateTime,2017-04-06 16:03:00  
8314090,VerifiedDateTime,2017-04-06 16:03:00  
8314090,VistaCreateDate,2017-04-06 17:55:10  
8314090,VistaEditDate,2017-04-06 17:55:10





# Our radiology dataset includes:

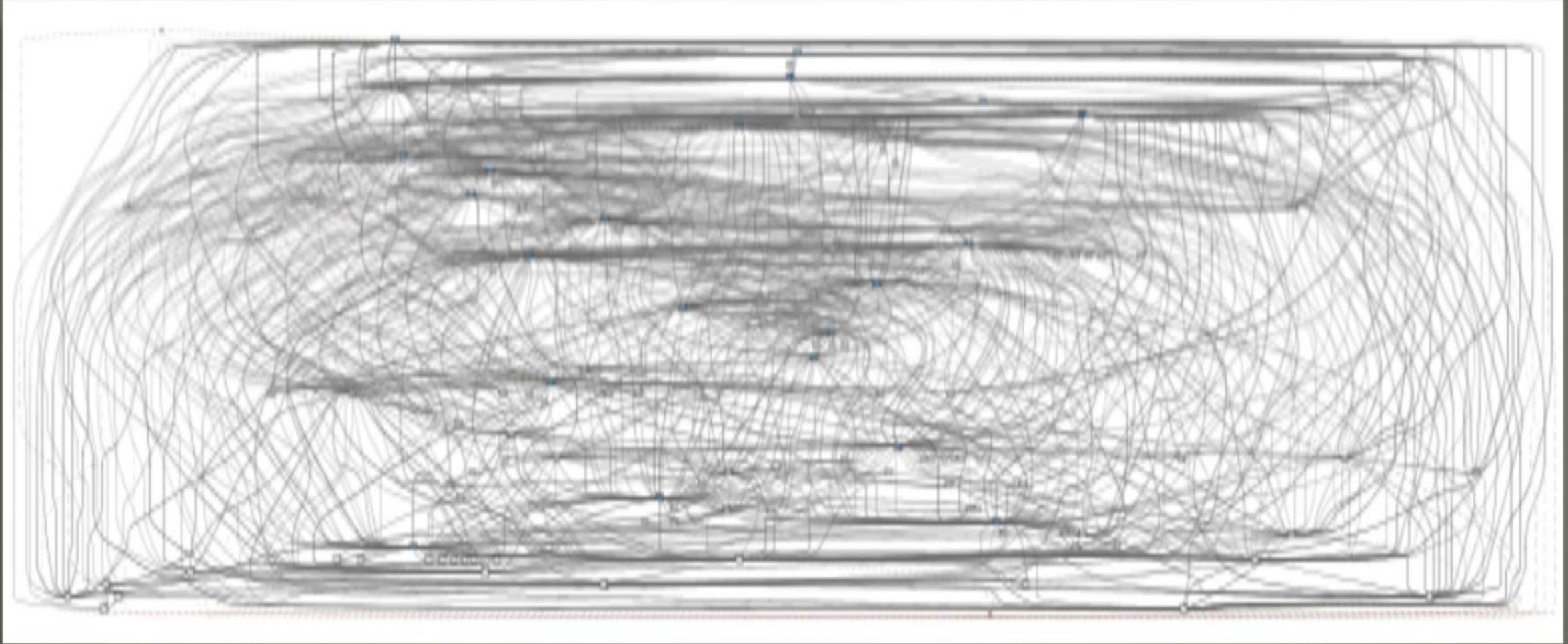
**860K cases**

**335K variants**

**18.5 million of events**

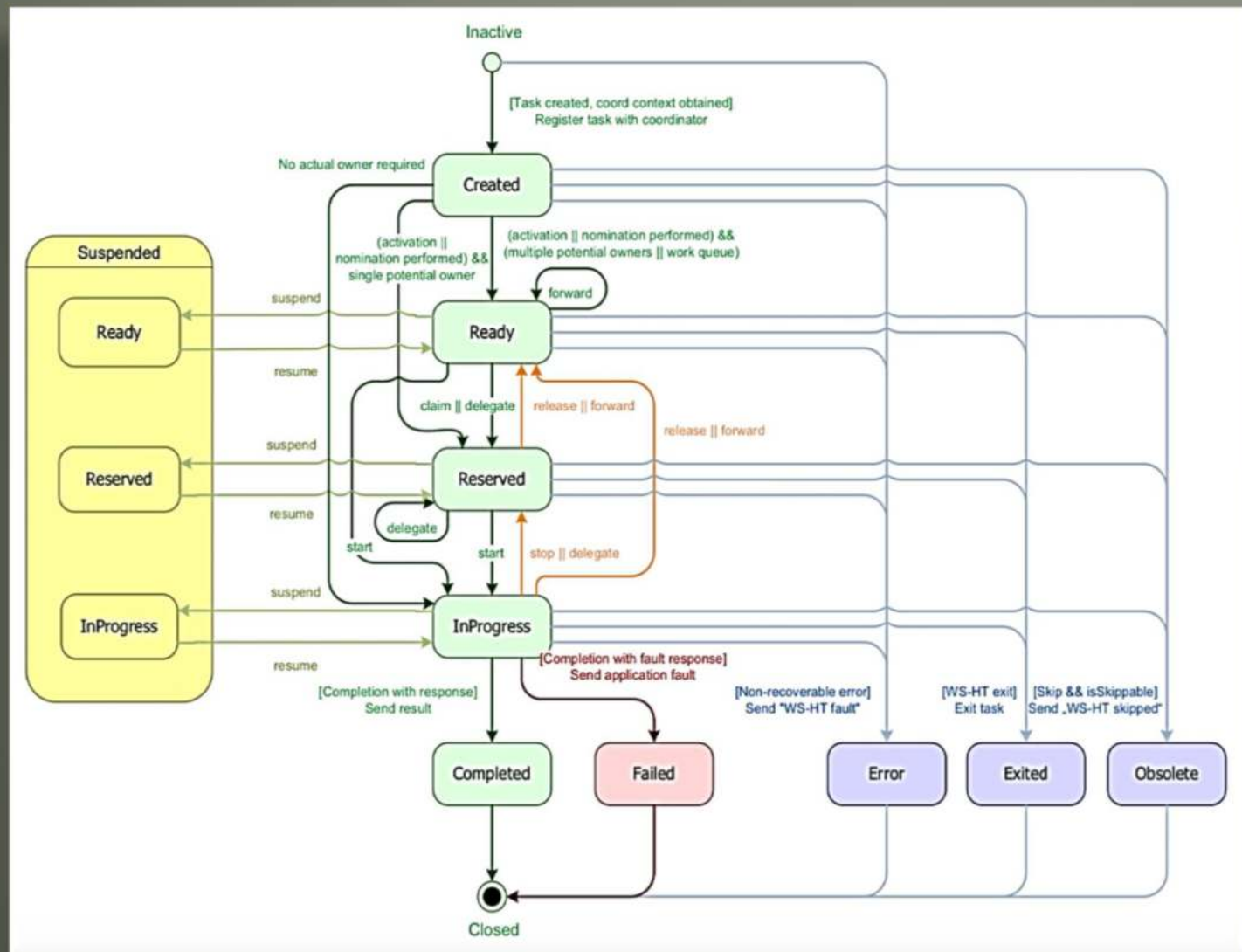
**Datasets like this will take months to just study one case!!**

The resulting process map is complex...



# OASIS WS-Human Task Specification – State Transition Diagram

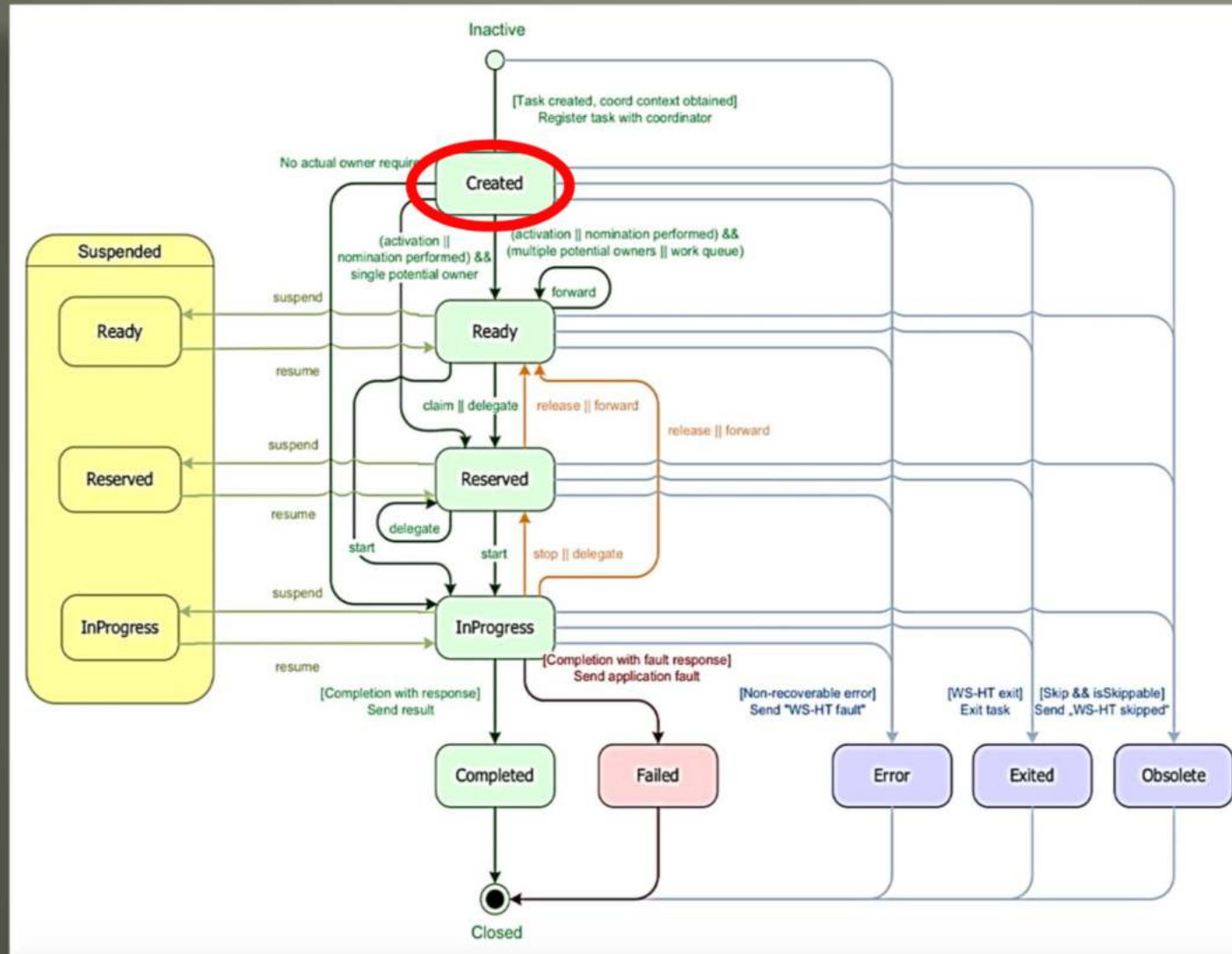
Human Task Behavior and State transitions Diagram from OASIS, *Web Services Human Task (WS-Human Task) Specification Version 1.1 - Committee Specification Draft 12 / Public Review Draft 05. 2012*, page 58.





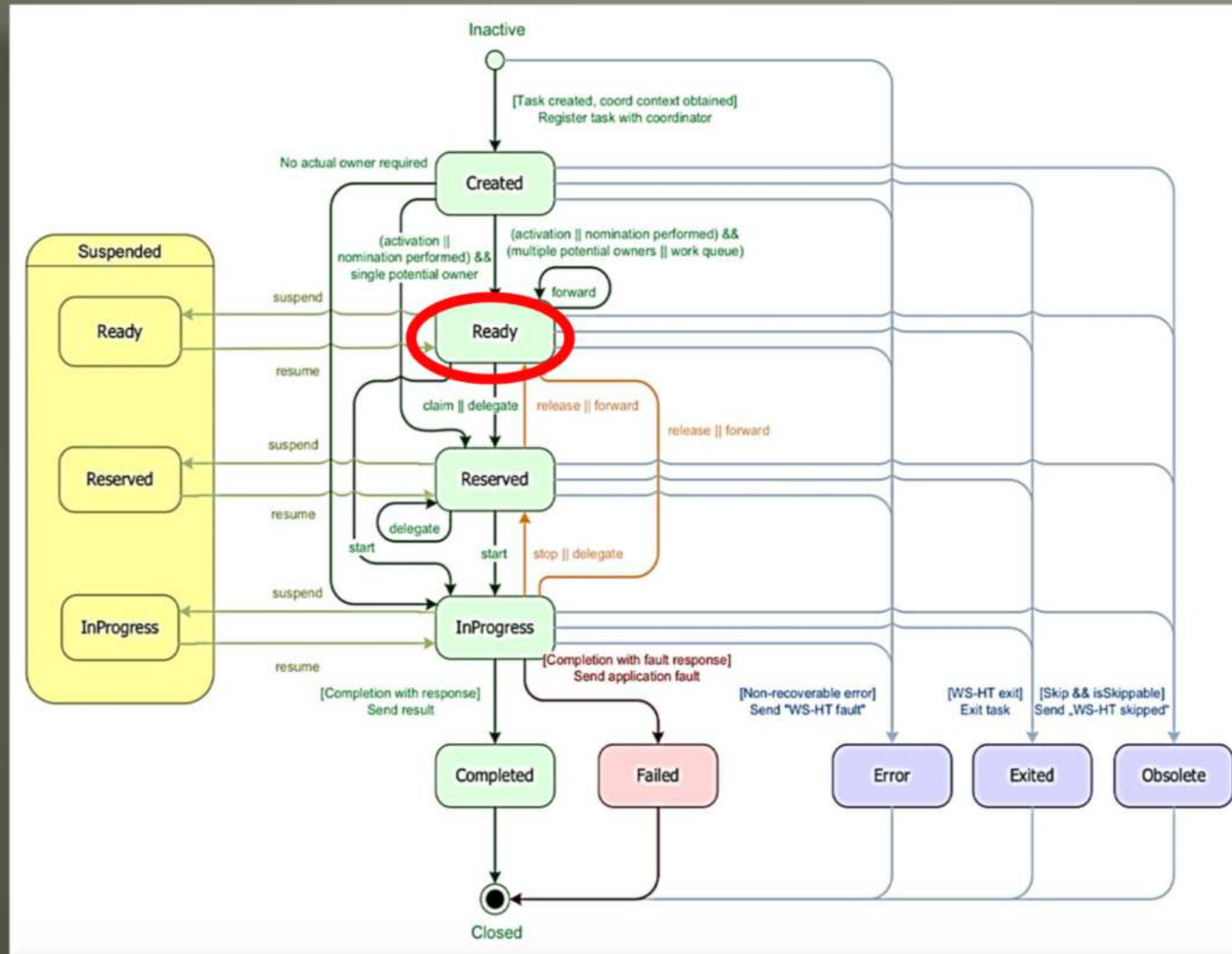
# OASIS WS-Human Task Specification – State Transition Diagram

Human Task Behavior and State transitions Diagram from OASIS, *Web Services Human Task (WS-Human Task) Specification Version 1.1 - Committee Specification Draft 12 / Public Review Draft 05. 2012, page 58.*



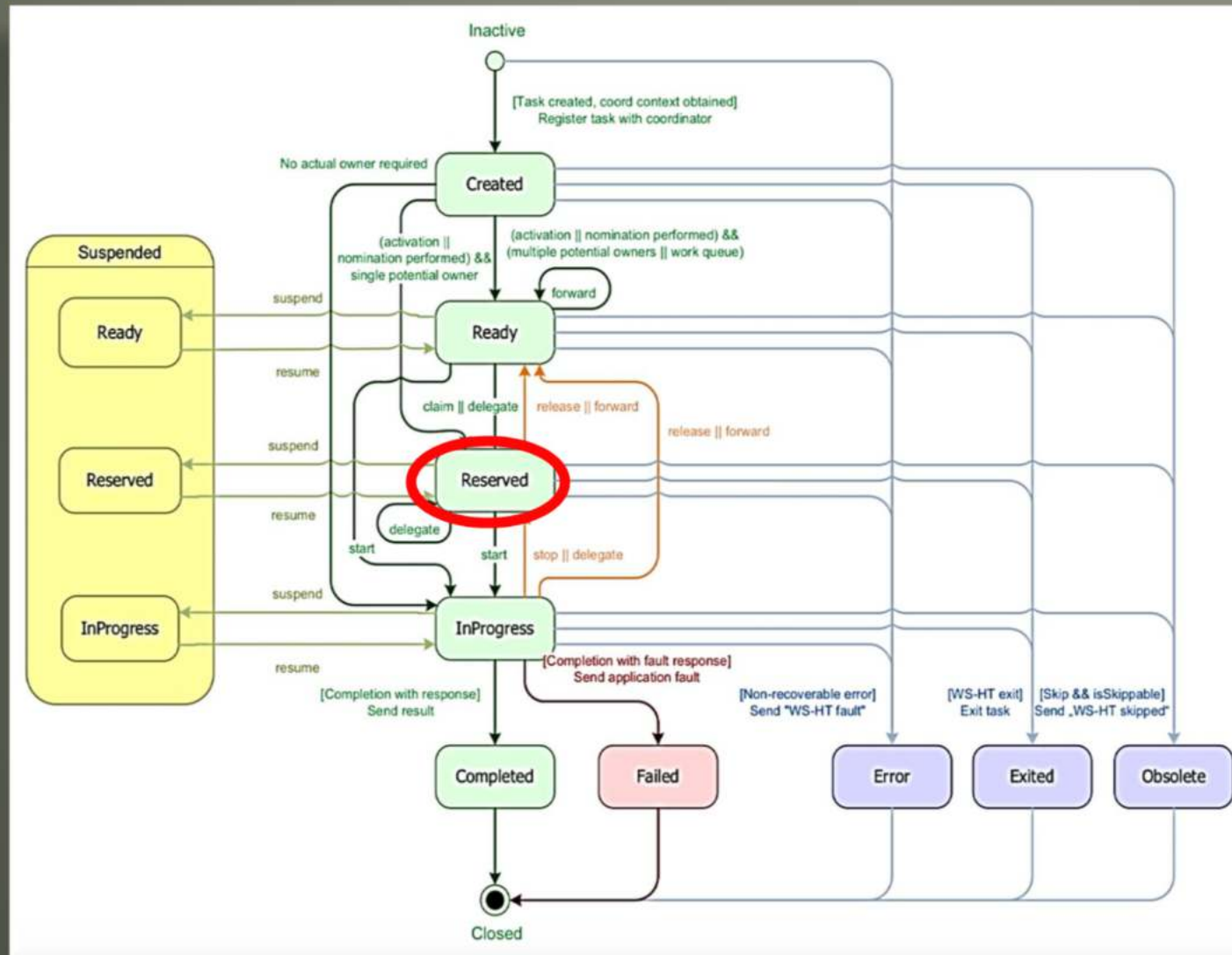
# OASIS WS-Human Task Specification – State Transition Diagram

Human Task Behavior and State transitions Diagram from OASIS, *Web Services Human Task (WS-Human Task) Specification Version 1.1 - Committee Specification Draft 12 / Public Review Draft 05. 2012*, page 58.



# OASIS WS-Human Task Specification – State Transition Diagram

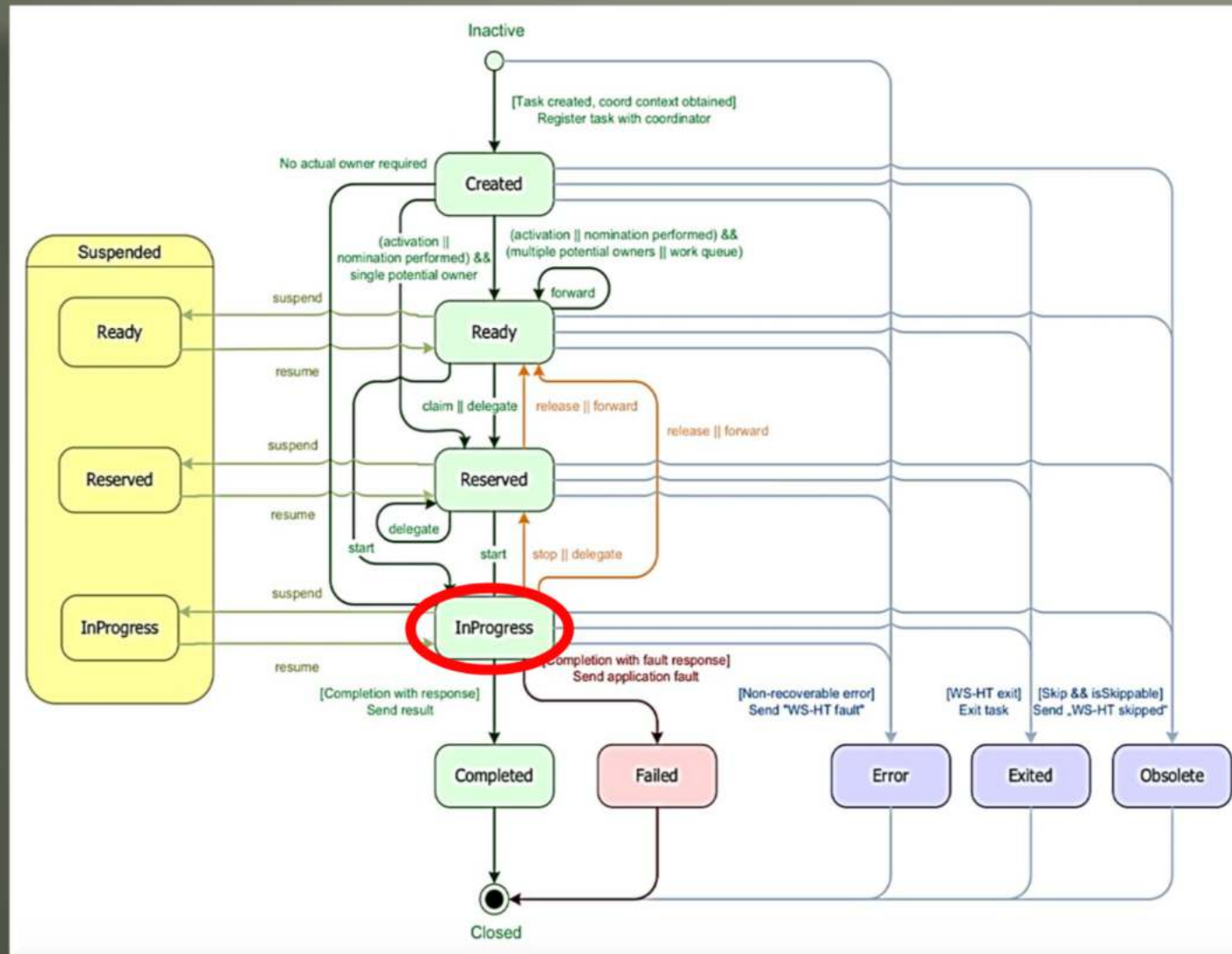
Human Task Behavior and State transitions Diagram from OASIS, *Web Services Human Task (WS-Human Task) Specification Version 1.1 - Committee Specification Draft 12 / Public Review Draft 05. 2012*, page 58.





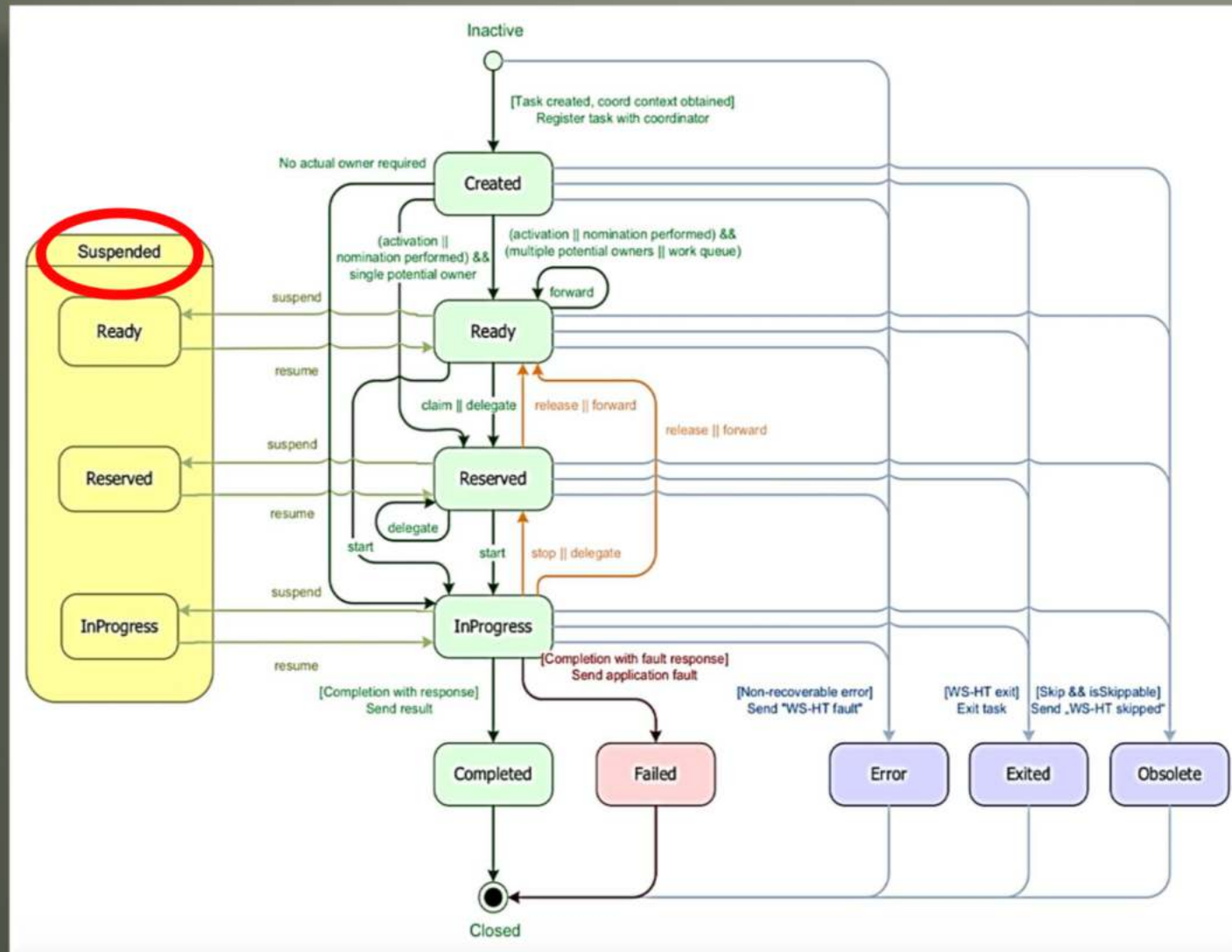
# OASIS WS-Human Task Specification – State Transition Diagram

Human Task Behavior and State transitions Diagram from OASIS, *Web Services Human Task (WS-Human Task) Specification Version 1.1 - Committee Specification Draft 12 / Public Review Draft 05. 2012, page 58.*



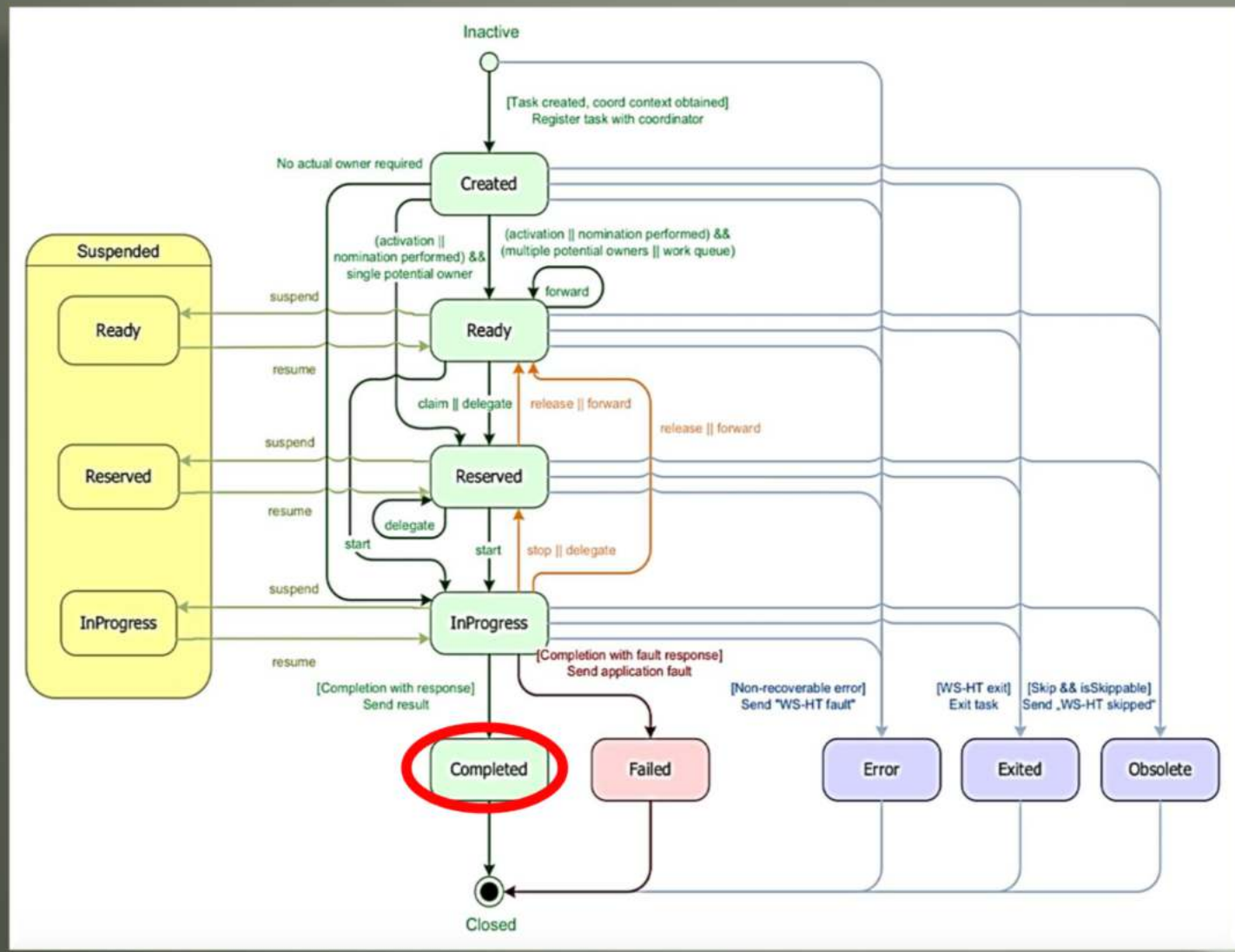
# OASIS WS-Human Task Specification – State Transition Diagram

Human Task Behavior and State transitions Diagram from OASIS, *Web Services Human Task (WS-Human Task) Specification Version 1.1 - Committee Specification Draft 12 / Public Review Draft 05. 2012*, page 58.



# OASIS WS-Human Task Specification – State Transition Diagram

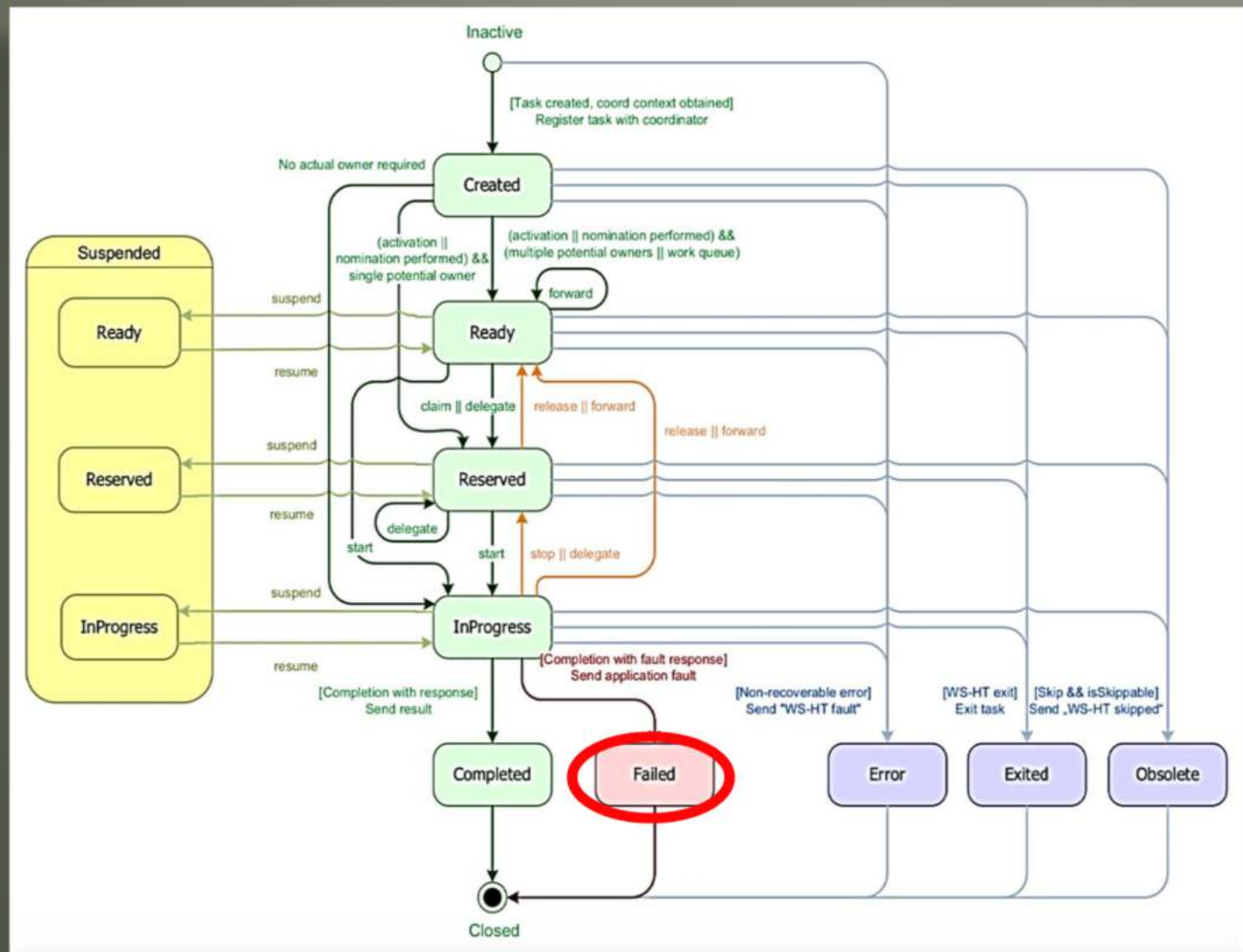
Human Task Behavior and State transitions Diagram from OASIS, *Web Services Human Task (WS-Human Task) Specification Version 1.1 - Committee Specification Draft 12 / Public Review Draft 05. 2012, page 58.*





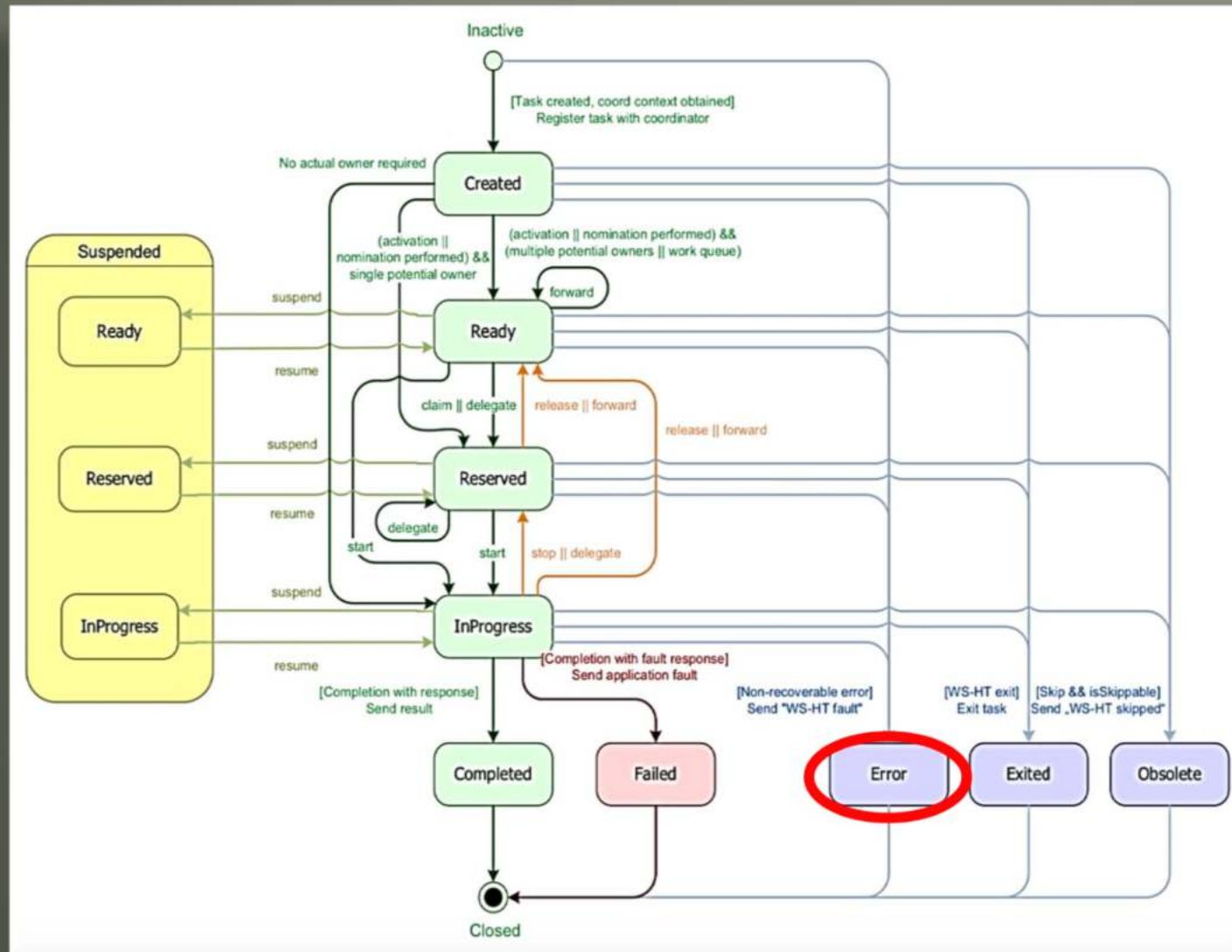
# OASIS WS-Human Task Specification – State Transition Diagram

Human Task Behavior and State transitions Diagram from OASIS, *Web Services Human Task (WS-Human Task) Specification Version 1.1 - Committee Specification Draft 12 / Public Review Draft 05. 2012, page 58.*



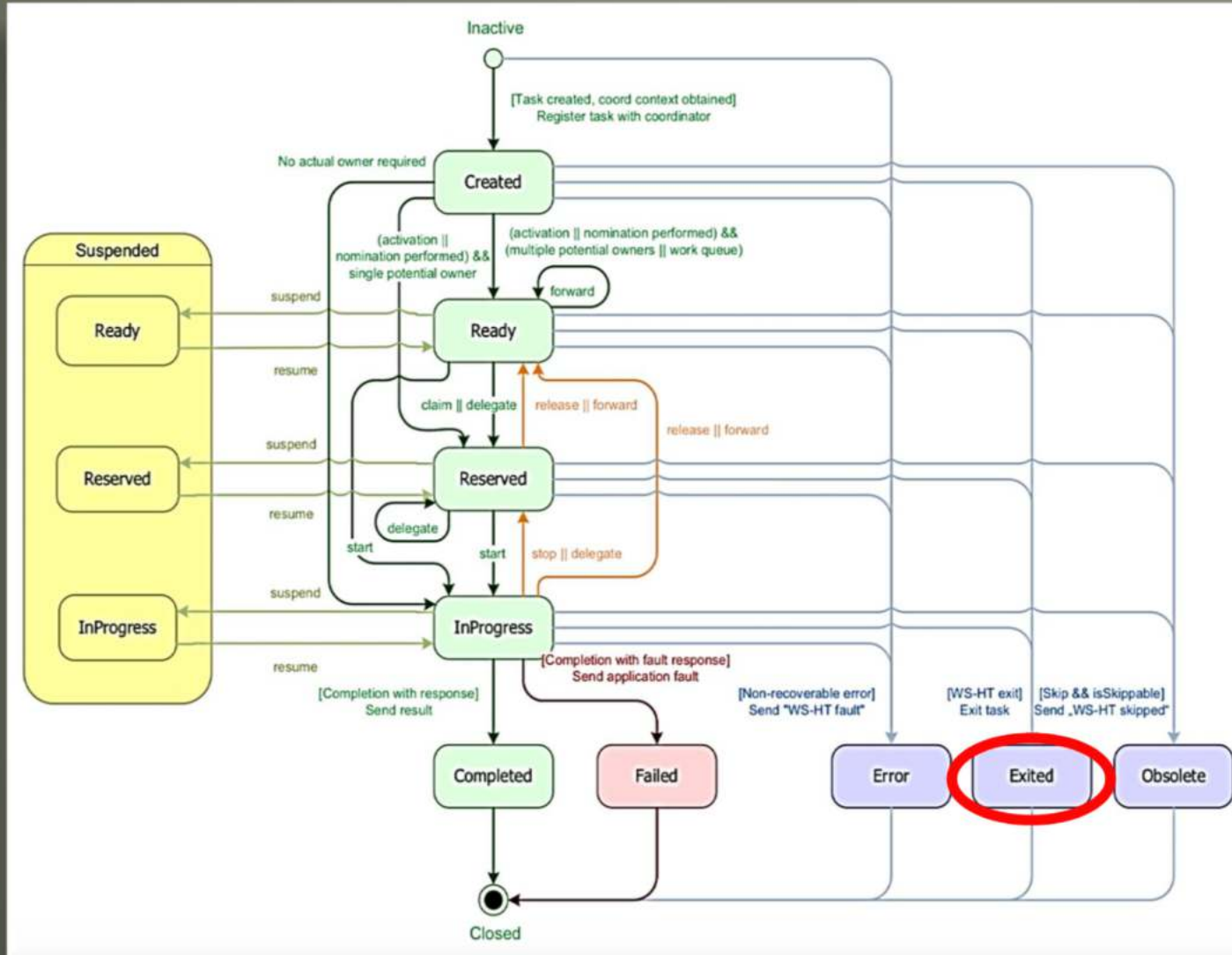
# OASIS WS-Human Task Specification – State Transition Diagram

Human Task Behavior and State transitions Diagram from OASIS, *Web Services Human Task (WS-Human Task) Specification Version 1.1 - Committee Specification Draft 12 / Public Review Draft 05. 2012, page 58.*



# OASIS WS-Human Task Specification – State Transition Diagram

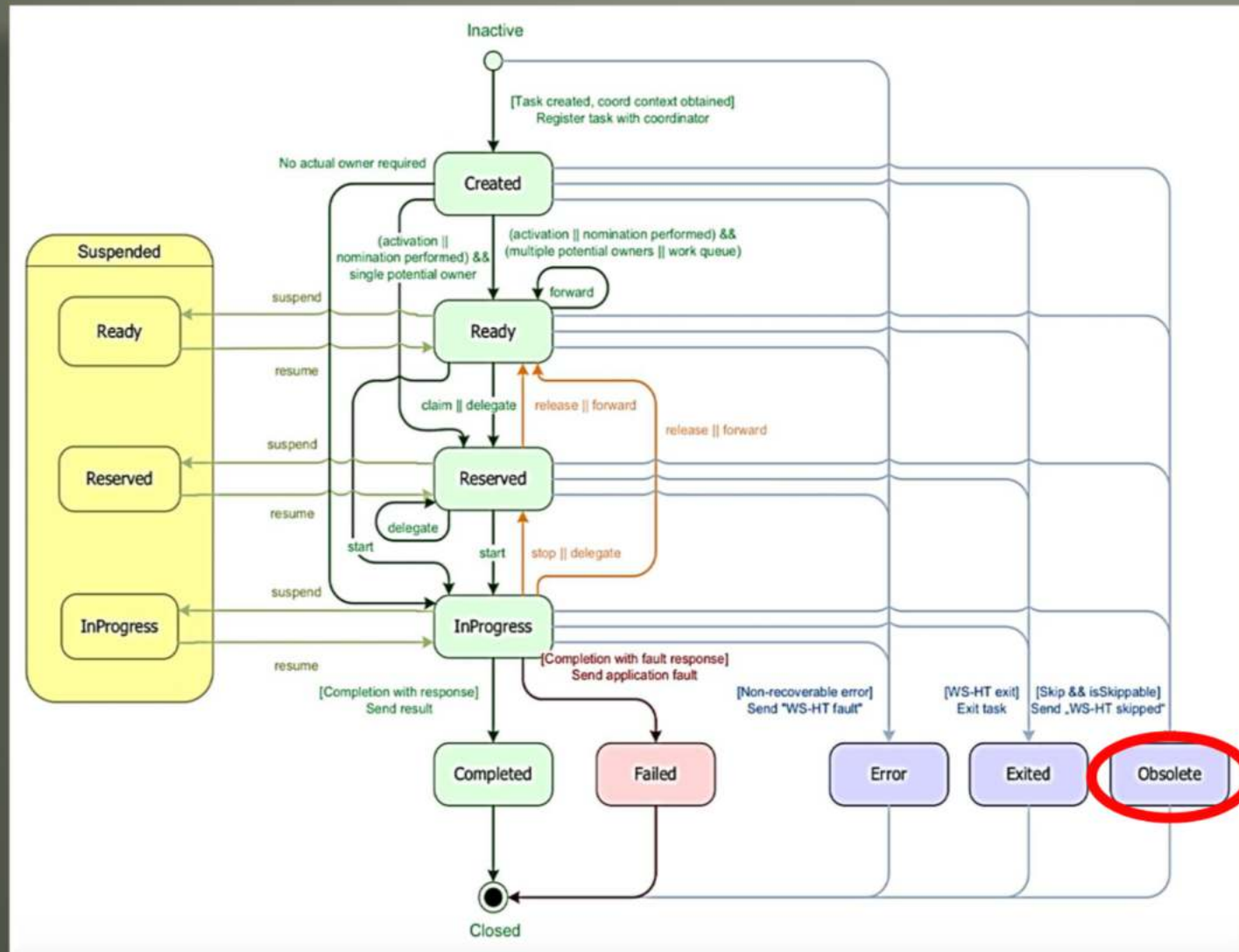
Human Task Behavior and State transitions Diagram from OASIS, *Web Services Human Task (WS-Human Task) Specification Version 1.1 - Committee Specification Draft 12 / Public Review Draft 05. 2012, page 58.*





# OASIS WS-Human Task Specification – State Transition Diagram

Human Task Behavior and State transitions Diagram from OASIS, *Web Services Human Task (WS-Human Task) Specification Version 1.1 - Committee Specification Draft 12 / Public Review Draft 05. 2012, page 58.*



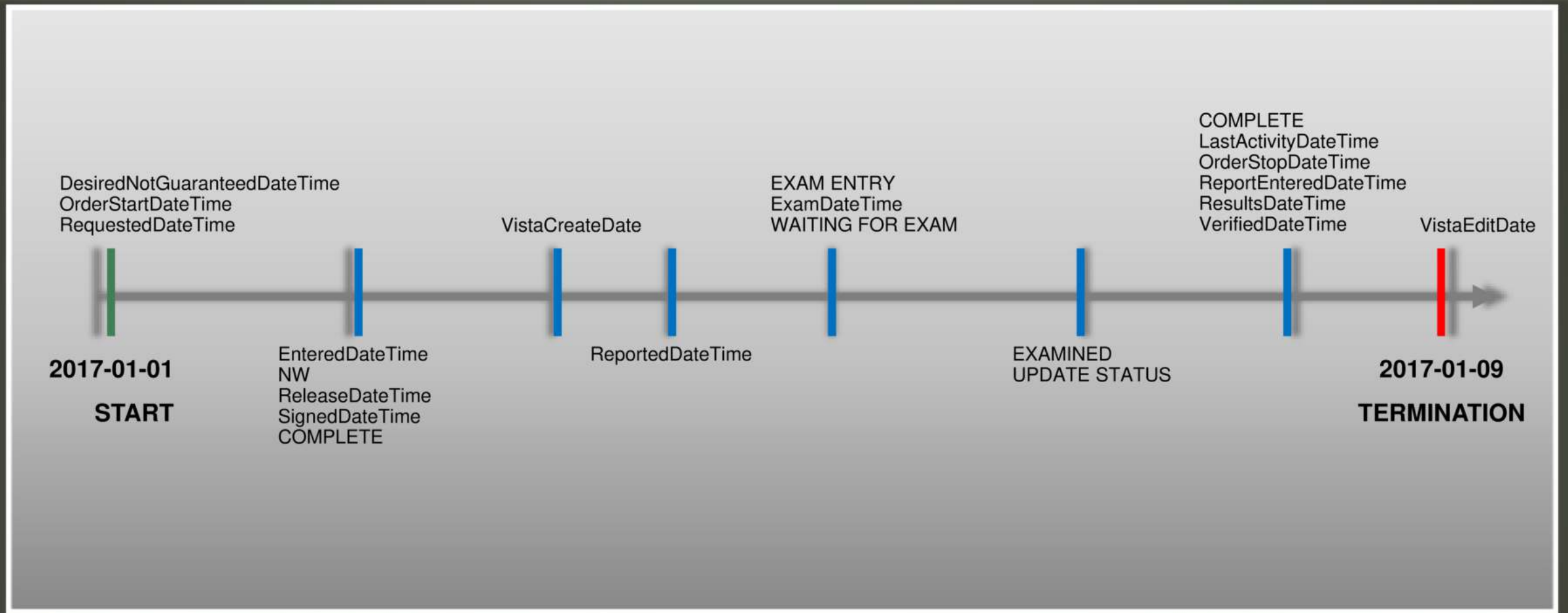


Figure 1. Temporal clustering of a sample order illustrated on a timeline.

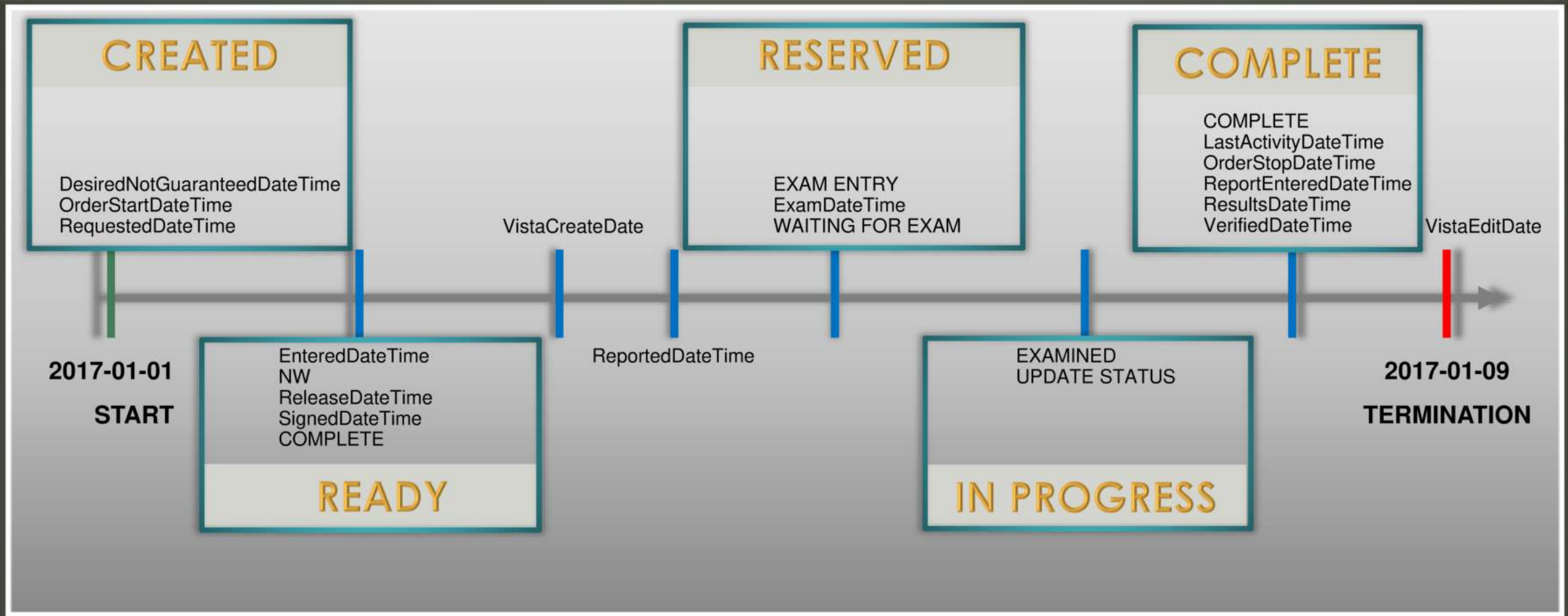


Figure 1. Temporal clustering of a sample order illustrated on a timeline.



# Radiology Order Sequence - **Complete** Example

## RAW EVENT SEQUENCE

8314090,DesiredNotGuaranteedDateTime,2017-04-06 00:00:00  
8314090,OrderStartDateTime,2017-04-06 00:00:00  
8314090,ReportedDateTime,2017-04-06 00:00:00  
8314090,RequestedDateTime,2017-04-06 00:00:00  
8314090,EnteredDateTime,2017-04-06 15:37:00  
8314090,NW,2017-04-06 15:37:00  
8314090,ReleaseDateTime,2017-04-06 15:37:00  
8314090,SignedDateTime,2017-04-06 15:37:00  
8314090,COMPLETE,2017-04-06 15:37:52  
8314090,EXAM ENTRY,2017-04-06 15:42:00  
8314090,ExamDateTime,2017-04-06 15:42:00  
8314090,WAITING FOR EXAM,2017-04-06 15:42:00  
8314090,EDIT BY CASE NO.,2017-04-06 15:47:00  
8314090,EXAMINED,2017-04-06 15:47:00  
8314090,COMPLETE,2017-04-06 16:03:00  
8314090,LastActivityDateTime,2017-04-06 16:03:00  
8314090,OrderStopDateTime,2017-04-06 16:03:00  
8314090,ReportEnteredDateTime,2017-04-06 16:03:00

## STATE TRANSITIONS

8314090,Created, 2017-04-06 15:37:00  
8314090,Ready, 2017-04-06 15:37:00  
8314090,Reserved, 2017-04-06 15:42:00  
8314090,InProgress, 2017-04-06 15:47:00  
8314090,Completed, 2017-04-06 16:03:00

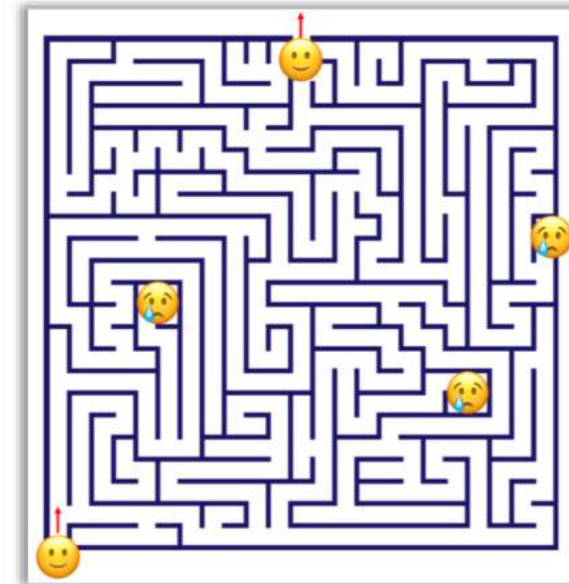
# Radiology Order Sequence - **Failed** Example

## RAW EVENT SEQUENCE

10560866,DesiredNotGuaranteedDateTime,2017-06-23 00:00:00  
10560866,OrderStartDateTime,2017-06-23 00:00:00  
10560866,RequestedDateTime,2017-06-23 00:00:00  
10560866,ExamDateTime,2017-06-23 09:57:00  
10560866,PastVisitDateTime,2017-06-23 09:57:00  
10560866,WAITING FOR EXAM,2017-06-23 09:57:00  
10560866,EnteredDateTime,2017-06-23 09:59:00  
10560866,EXAM ENTRY,2017-06-23 09:59:00  
10560866,NW,2017-06-23 09:59:00  
10560866,ReleaseDateTime,2017-06-23 09:59:00  
10560866,WAITING FOR EXAM,2017-06-23 09:59:00  
10560866,DISCONTINUED,2017-06-23 09:59:33  
...  
10560866,COMPLETE,2017-06-23 11:42:00  
10560866,ResultsDateTime,2017-06-23 11:42:00  
10560866,EXAMINED,2017-06-23 14:41:00  
10560866,CANCELLED,2017-06-23 15:03:00  
10560866,DiscontinuedDateTime,2017-06-23 15:03:00  
...

## STATE TRANSITIONS

10560866,Reserved, 2017-06-23 09:57:00  
10560866,Created, 2017-06-23 09:59:00  
10560866,Ready, 2017-06-23 09:59:00  
10560866,Completed, 2017-06-23 11:42:00  
10560866,Failed, 2017-06-23 15:03:00





## Mapping rules that generate simplified event logs from raw event sequence data.

Rule/s	Transition to
If Entered Date Time is found, transition to Created.	Created
If Released Date Time and a new (NW) action are taken, transition to Ready. If Released Date Time and a release hold (RL) action are taken, and order is at Suspended, transition to Ready. If Released Date Time and Signed Date Time are found, and order is not at Ready and no Order Stop Date Time is recorded, transition to Ready.	Ready
If Released Date Time and a hold (HD) action are taken, transition to Suspended.	Suspended
If EXAM ENTRY is found, transition to Reserved.	Reserved
If there is any cluster of Dates or Actions take place after transitioning to Reserved, transition to InProgress.	InProgress
If there is Results Date Time, transition to Completed.	Completed
If order is at Completed and there is Discontinued Date Time recorded, then transition to Failed.	Failed
If order is at Ready and there is Discontinued Date Time recorded, then transition to Error.	Error
If there is Discontinued Date Time recorded (else: not at Completed or Ready), then transition to Exited.	Exited



# Horizontal sequence example

```
8314090 |DesiredNotGuaranteedDateTime,2017-04-06 00:00:00|
ReportedDateTime,2017-04-06 00:00:00|RequestedDateTime,2017-04-06 00:00:00|
OrderStartDateTime,2017-04-06 00:00:00|SignedDateTime,2017-04-06 15:37:00|
EnteredDateTime,2017-04-06 15:37:00|OrderActionDateTime,2017-04-06 15:37:00|
ReleaseDateTime,2017-04-06 15:37:00|RequestEnteredDateTime,2017-04-06 15:37:52|
WAITING FOR EXAM,2017-04-06 15:42:00|StatusChangeDateTime,2017-04-06 15:42:00|
ExamDateTime,2017-04-06 15:42:00|EXAM ENTRY,2017-04-06 15:42:00|
StatusChangeDateTime,2017-04-06 15:47:00|EXAMINED,2017-04-06 15:47:00|
EDIT BY CASE NO.,2017-04-06 15:47:00|VerifiedDateTime,2017-04-06 16:03:00|
ReportEnteredDateTime,2017-04-06 16:03:00|
LastActivityDateTime,2017-04-06 16:03:00|OrderStopDateTime,2017-04-06 16:03:00|
COMPLETE,2017-04-06 16:03:00|StatusChangeDateTime,2017-04-06 16:03:00|
ResultsDateTime,2017-04-06 16:03:00|VistaCreateDate,2017-04-06 17:55:10|
VistaEditDate,2017-04-06 17:55:10
```

If **“Entered Date Time”** is found, transition to **“Created”**

```
if "EnteredDateTime" in evList:  
    i = evList.index("EnteredDateTime")  
    state = "Created"  
    pair = (state, tp[i][1])  
    transitions.append(pair)
```

Code developed by Ozmen Ozgur

# Horizontal sequence example

CREATED

```
8314090 | DesiredNotGuaranteedDateTime,2017-04-06 00:00:00 |
ReportedDateTime,2017-04-06 00:00:00 | RequestedDateTime,2017-04-06 00:00:00 |
OrderStartDateTime,2017-04-06 00:00:00 | SignedDateTime,2017-04-06 15:37:00 |
EnteredDateTime,2017-04-06 15:37:00 | OrderActionDateTime,2017-04-06 15:37:00 |
ReleaseDateTime,2017-04-06 15:37:00 | RequestEnteredDateTime,2017-04-06 15:37:52 |
WAITING FOR EXAM,2017-04-06 15:42:00 | StatusChangeDateTime,2017-04-06 15:42:00 |
ExamDateTime,2017-04-06 15:42:00 | EXAM ENTRY,2017-04-06 15:42:00 |
StatusChangeDateTime,2017-04-06 15:47:00 | EXAMINED,2017-04-06 15:47:00 |
EDIT BY CASE NO.,2017-04-06 15:47:00 | VerifiedDateTime,2017-04-06 16:03:00 |
ReportEnteredDateTime,2017-04-06 16:03:00 |
LastActivityDateTime,2017-04-06 16:03:00 | OrderStopDateTime,2017-04-06 16:03:00 |
COMPLETE,2017-04-06 16:03:00 | StatusChangeDateTime,2017-04-06 16:03:00 |
ResultsDateTime,2017-04-06 16:03:00 | VistaCreateDate,2017-04-06 17:55:10 |
VistaEditDate,2017-04-06 17:55:10
```



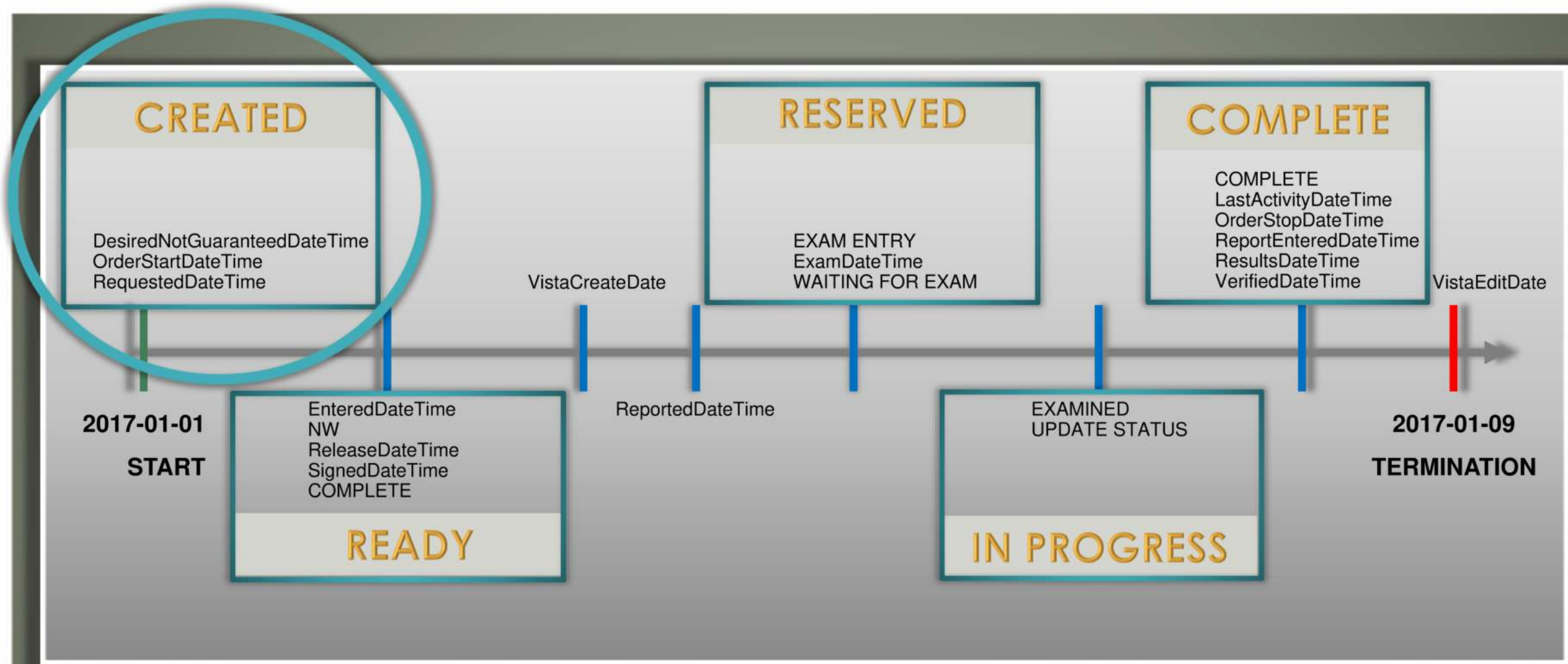


Figure 1. Temporal clustering of a sample order illustrated on a timeline.

However, identification of transitions  
to other states  
was not as straightforward

If **“Released Date Time”** and a new (NW) action are taken, transition to **“Ready”**.

If **“Released Date Time”** and a release hold (RL) action are taken, and order is at Suspended, transition to **“Ready”**.

```
if "ReleaseDateTime" in evList and 'NW' in evList and state != 'Ready':  
    i = evList.index("NW")  
    state = "Ready"  
    pair = (state, tp[i][1])  
    transitions.append(pair)  
if state == "Suspended":  
    if "ReleaseDateTime" in evList and 'RL' in evList:  
        i = evList.index("RL")  
        state = "Ready"  
        pair = (state, tp[i][1])  
        transitions.append(pair)
```



If **“Released Date Time”** and **“Signed Date Time”** are found, and order is not at **“Ready”** and no **“Order Stop Date Time”** is recorded, transition to **“Ready”**.

```
if "ReleaseDateTime" in evList and "OrderStopDateTime" not in evList and "SignedDateTime" in evList and state != 'Ready':  
    i = evList.index("ReleaseDateTime")  
    state = "Ready"  
    pair = (state, tp[i][1])  
    transitions.append(pair)
```

# Horizontal sequence example

```
8314090 | DesiredNotGuaranteedDateTime, 2017-04-06 00:00:00 |
ReportedDateTime, 2017-04-06 00:00:00 | RequestedDateTime, 2017-04-06 00:00:00 |
OrderStartDateTime, 2017-04-06 00:00:00 | SignedDateTime, 2017-04-06 15:37:00 |
EnteredDateTime, 2017-04-06 15:37:00 | OrderActionDateTime, 2017-04-06 15:37:00 |
ReleaseDateTime, 2017-04-06 15:37:00 | RequestEnteredDateTime, 2017-04-06 15:37:52 |
WAITING FOR EXAM, 2017-04-06 15:42:00 | StatusChangeDateTime, 2017-04-06 15:42:00 |
ExamDateTime, 2017-04-06 15:42:00 | EXAM ENTRY, 2017-04-06 15:42:00 |
StatusChangeDateTime, 2017-04-06 15:47:00 | EXAMINED, 2017-04-06 15:47:00 |
EDIT BY CASE NO., 2017-04-06 15:47:00 | VerifiedDateTime, 2017-04-06 16:03:00 |
ReportEnteredDateTime, 2017-04-06 16:03:00 |
LastActivityDateTime, 2017-04-06 16:03:00 | OrderStopDateTime, 2017-04-06 16:03:00 |
COMPLETE, 2017-04-06 16:03:00 | StatusChangeDateTime, 2017-04-06 16:03:00 |
ResultsDateTime, 2017-04-06 16:03:00 | VistaCreateDate, 2017-04-06 17:55:10 |
VistaEditDate, 2017-04-06 17:55:10
```

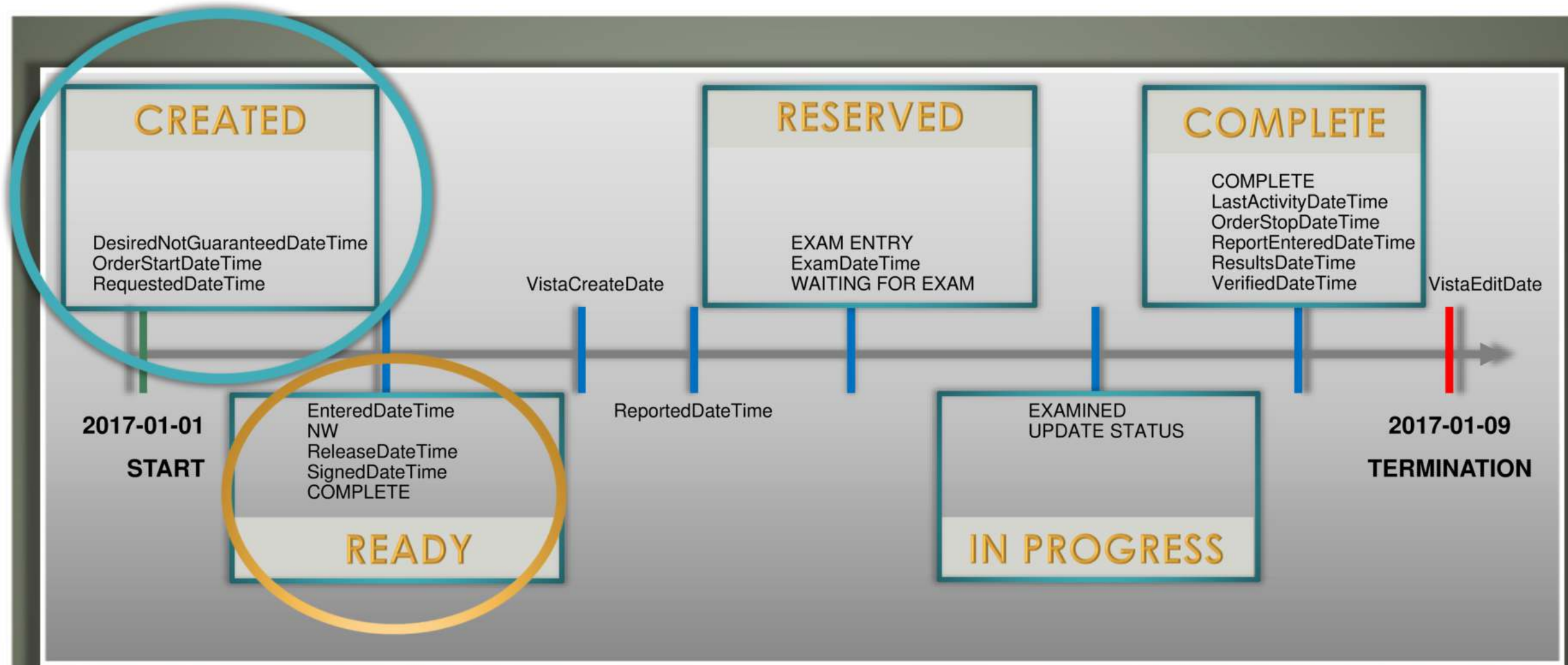


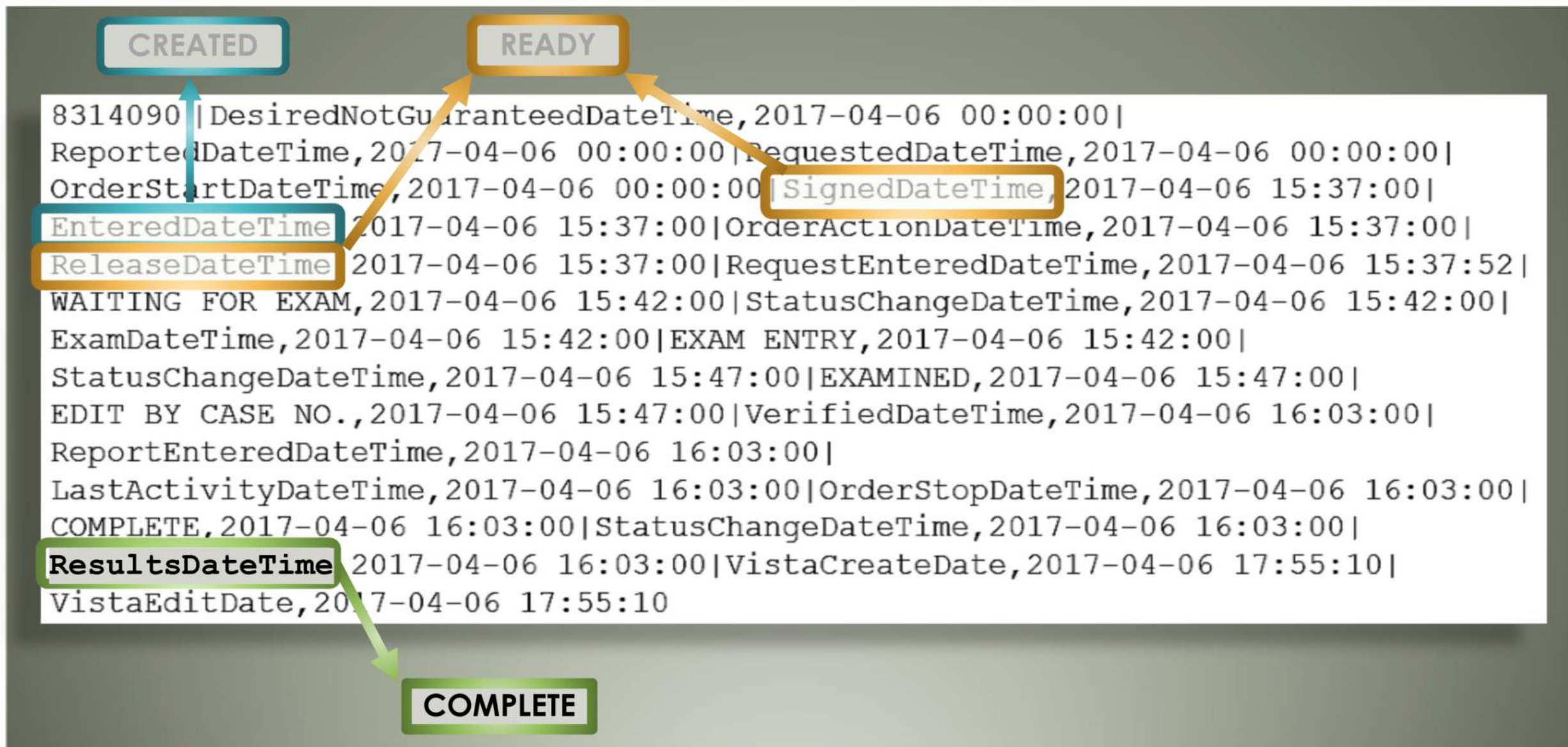
Figure 1. Temporal clustering of a sample order illustrated on a timeline.



If there is **“Results Date Time”**, transition to **“Completed”**

```
if “ResultsDateTime” in evList:  
    i = evList.index(“ResultsDateTime”)  
    state = “Completed”  
    pair = (state, tp[i][1])  
    transitions.append(pair)  
    continue
```

# Horizontal sequence example



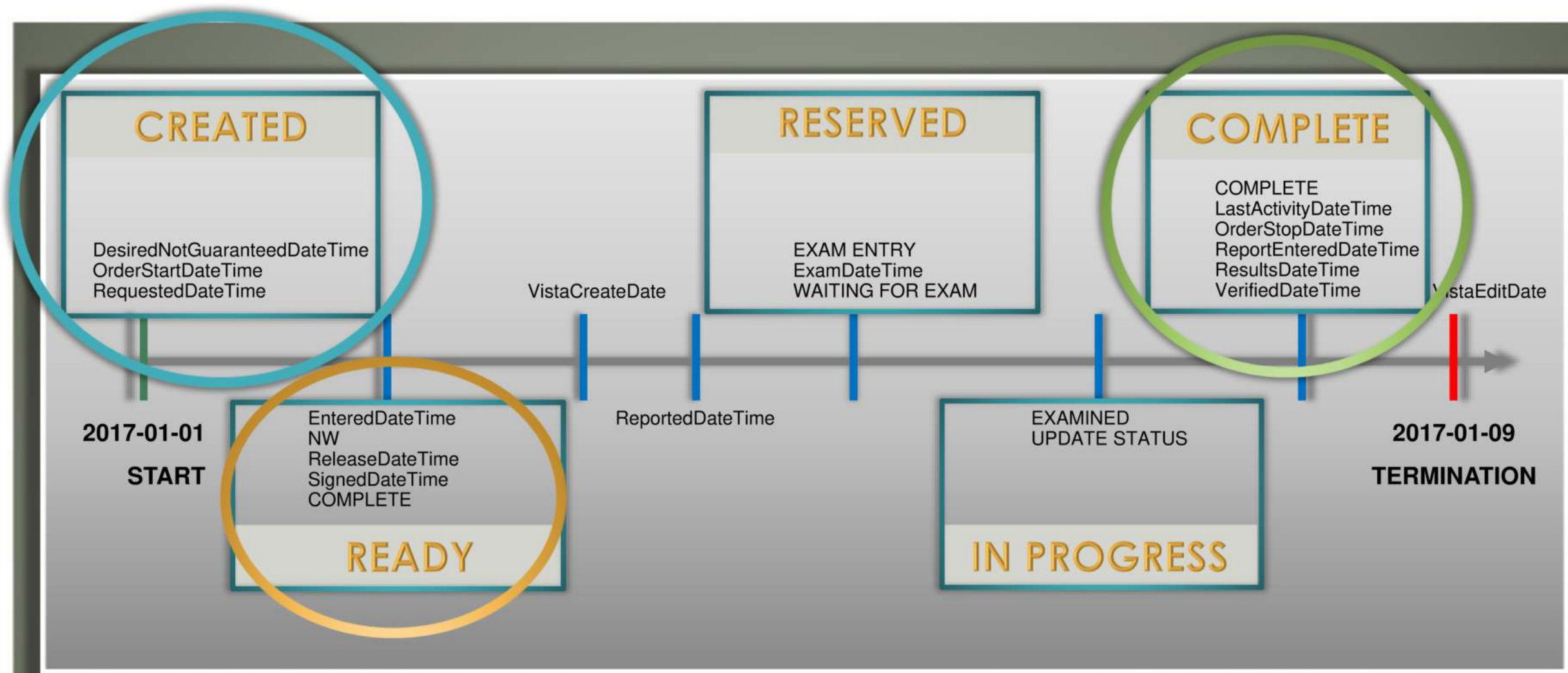


Figure 1. Temporal clustering of a sample order illustrated on a timeline.



- If order is at Ready and there is a **“Discontinued Date Time”** recorded, then transition to **“Error”**.
- If order is at **“Completed”** and there is a **“Discontinued Date Time”** recorded, then transition to **“Failed”**.
- If there is a **“Discontinued Date Time”** recorded (else: not at **“Completed”** or **“Ready”**), then transition to **“Exited”**.

```
if 'DiscontinuedDateTime' in evList:
    i = evList.index("DiscontinuedDateTime")
    if state == 'Created':
        state = "Error"
        pair = (state, tp[i][1])
        transitions.append(pair)
        continue
    elif state == 'Completed':
        state = "Failed"
        pair = (state, tp[i][1])
        transitions.append(pair)
        continue
    else:
        state = "Exited"
        pair = (state, tp[i][1])
        transitions.append(pair)
        continue
```

# Radiology Order Sequence - Failed Example

## RAW EVENT SEQUENCE

10560866,DesiredNotGuaranteedDateTime,2017-06-23 00:00:00  
10560866,OrderStartDateTime,2017-06-23 00:00:00  
10560866,RequestedDateTime,2017-06-23 00:00:00  
10560866,ExamDateTime,2017-06-23 09:57:00  
10560866,PastVisitDateTime,2017-06-23 09:57:00  
10560866,WAITING FOR EXAM,2017-06-23 09:57:00  
10560866,EnteredDateTime,2017-06-23 09:59:00  
10560866,EXAM ENTRY,2017-06-23 09:59:00  
10560866,NW,2017-06-23 09:59:00  
10560866,ReleaseDateTime,2017-06-23 09:59:00  
10560866,WAITING FOR EXAM,2017-06-23 09:59:00  
10560866,DISCONTINUED,2017-06-23 09:59:33  
...  
10560866,COMPLETE,2017-06-23 11:42:00  
10560866,ResultsDateTime,2017-06-23 11:42:00  
10560866,EXAMINED,2017-06-23 14:41:00  
10560866,CANCELLED,2017-06-23 15:03:00  
10560866,DiscontinuedDateTime,2017-06-23 15:03:00  
...

## STATE TRANSITIONS

10560866,Reserved, 2017-06-23 09:57:00  
10560866,Created, 2017-06-23 09:59:00  
10560866,Ready, 2017-06-23 09:59:00  
10560866,Completed, 2017-06-23 11:42:00  
10560866,Failed, 2017-06-23 15:03:00

FAILED

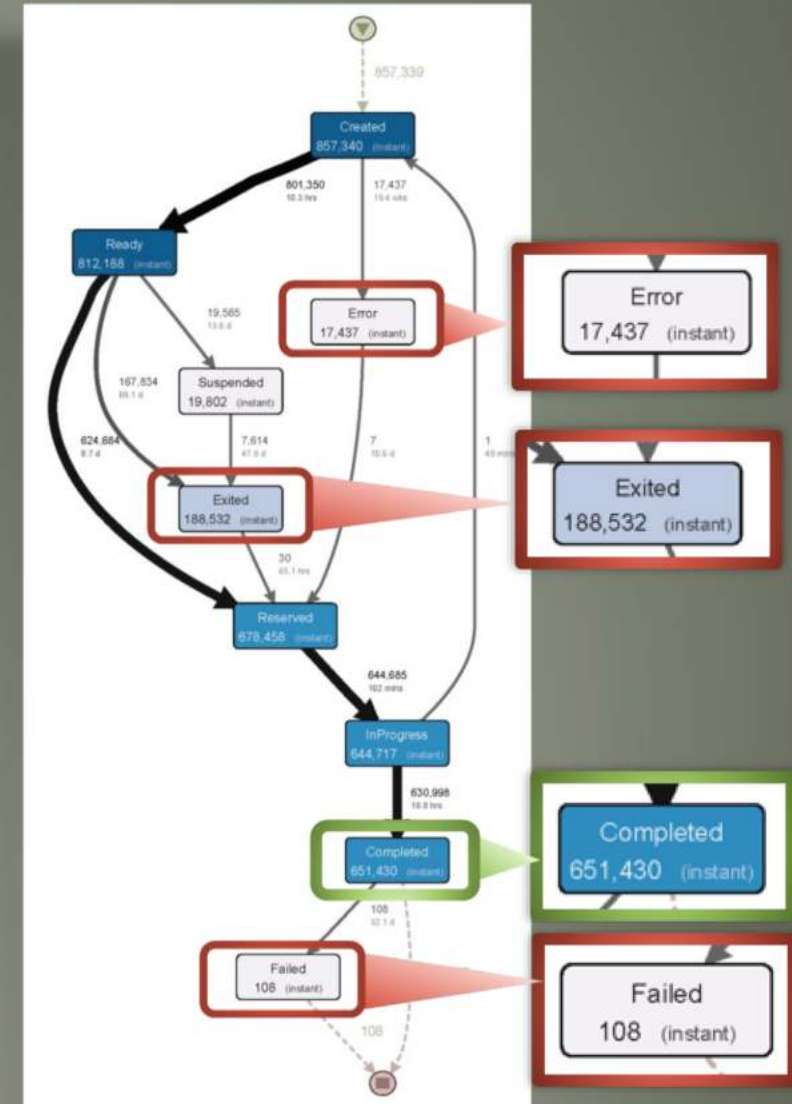


# Process Mining Diagram

## State transition diagram for radiology orders

Process Mining is used to discover the process that most radiology orders follow.

- The path that most sequences take is presented by the dark blue boxes and the black bold arrows (~76%).
- We also identified the sequences that did not complete successfully Error (~2%) / Exited (22%) / Failed(0.01%)

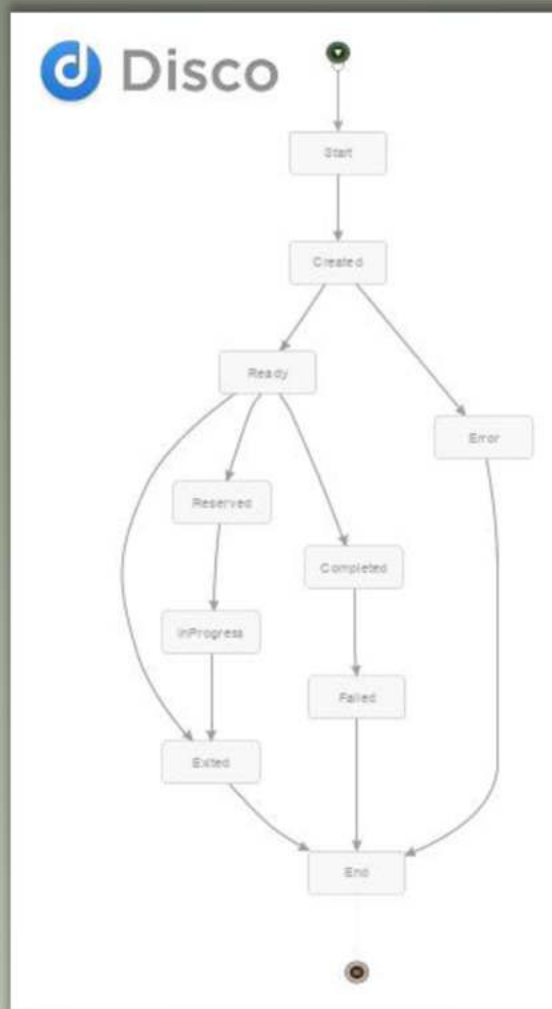




# Process Mining - Animation sample

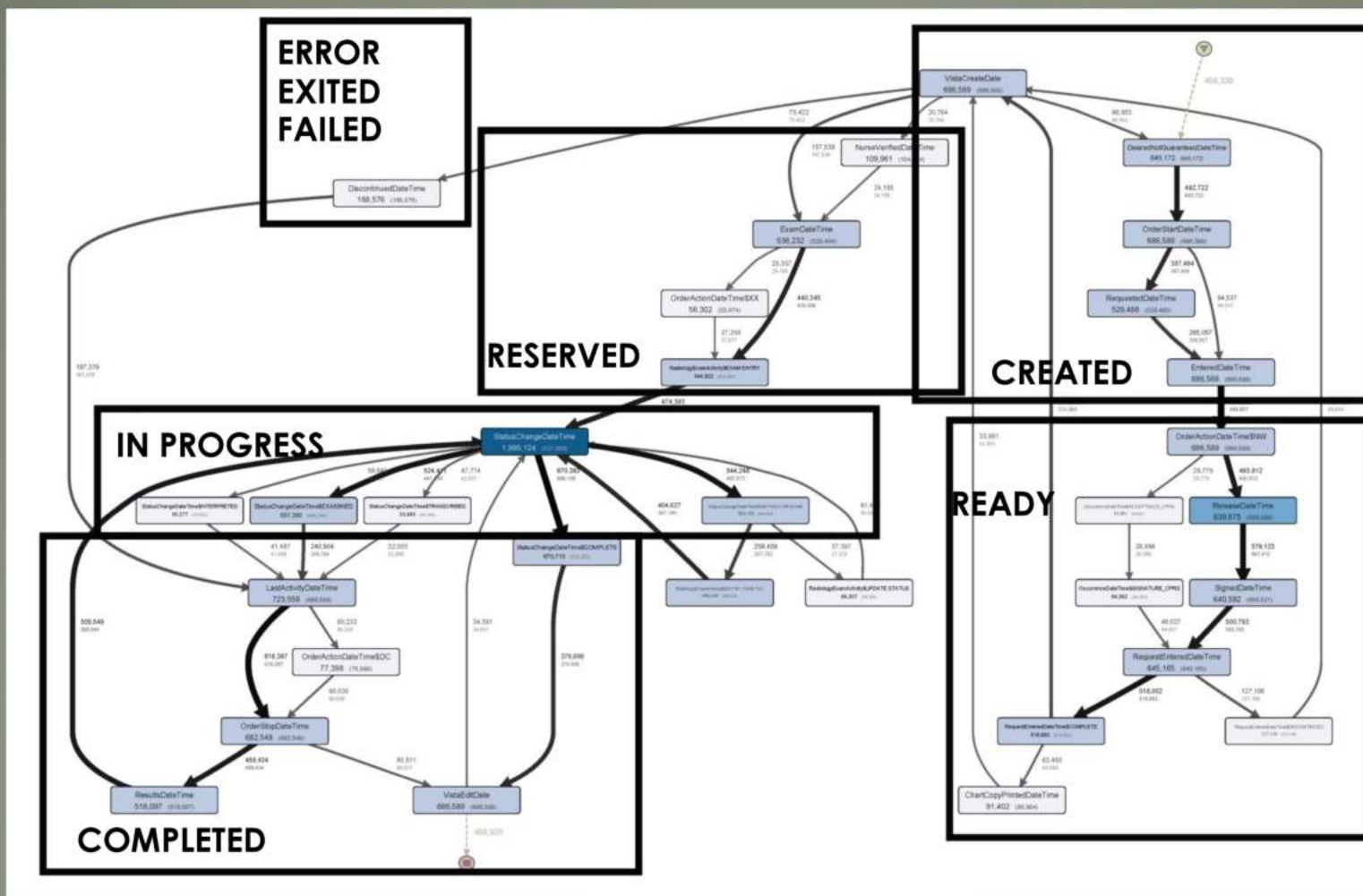
This animation presents filtered data in Disco to keep only cases that terminated in unsuccessful states:

- **Error**,
- **Exited**, and
- **Failed**



Identifying the unsuccessful cases is a major objective in our study.

# Process Mining – Radiology Process Model Map



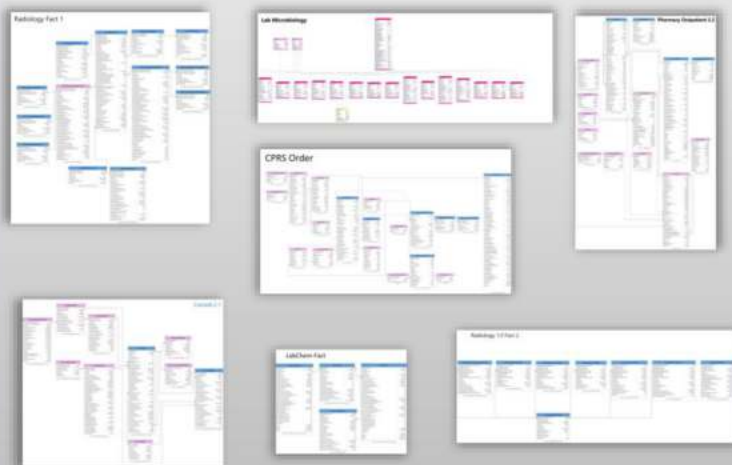
# Analysis Results

- We compared completed cases against failed, exited and error cases. We observed that completed cases:
  - Followed similar patterns
  - The activities mostly follow a sequence with few loops
  - Spent similar amounts of time in between steps
  - Times from start to end were regular
- In contrast, the failed, exited or error cases, often:
  - Use unusual activities
  - Include Zero Timestamps (i.e. dates far in the past or far in the future)
  - They are incomplete
  - They present extra loops

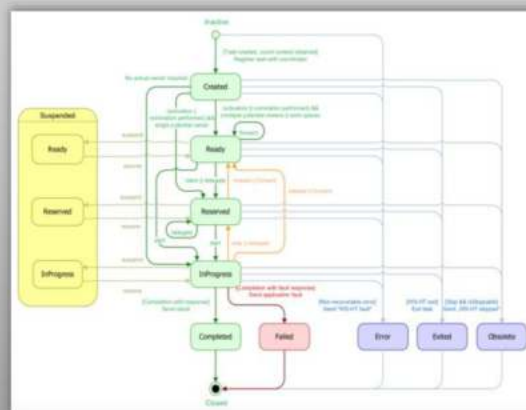
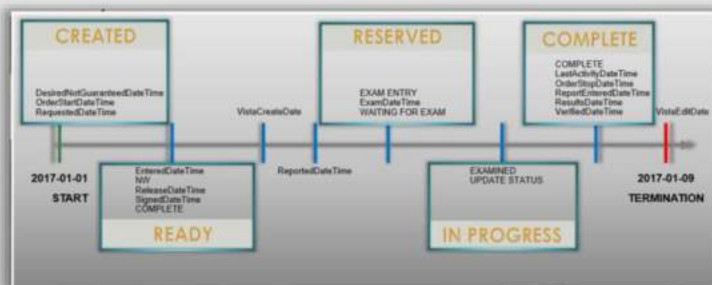


# Temporal clustering and mapping summary

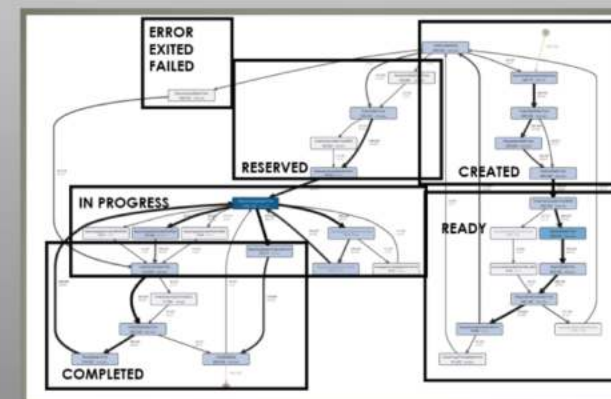
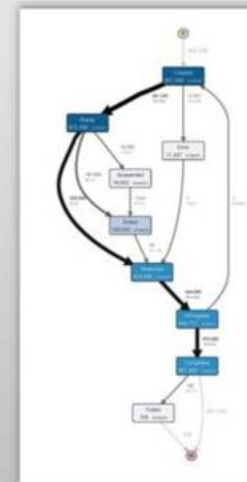
## Extract Features



## Temporal Clustering



## Process Mining



# Strategies to Reduce Complexity That Worked For Us

- Sorting event sequences not only by date but also by activity reduced the number of variants to about a quarter of the dataset.
- It is important to distinguish each feature with a distinct unique name, otherwise we may see spider like areas in the process maps from events with the same name but on different contexts, which create false loops.
- Mapping the dataset activities to the OASIS human task state transition diagram was extremely helpful to rapidly identify and visualize those cases that ended in error states, such as Failed, Exited, and Error.

# Data Quality Problems We Encountered

- Our study detected redundancy of data where some values refer to the same classification but are entered with different names. An example: complete vs Complete vs COMPLETE vs COMPLETED vs completed, vs COM.
- Daylight saving time can affect the analysis and impact process maps. Attention should be placed to activities recorded during daylight saving time.
- Our study detected Zero Timestamps [faulty timestamps that are far in the past (i.e. 1900) or far in the future (i.e. 2100)]. An example: an activity with date: 01.01.1900 affected the overall study and analysis by impacting durations, variants and process maps. Those cases were filtered in order to identify the real durations.



# Conclusion

- Next to the *Radiology* domain, we also applied process mining in the following data domains: *Consults*, *Lab Services*, *RxOut* (outpatient medications) and *CPRSOrder* (Computerized Patient Record System used for scheduling). An initial set of metrics and key performance indicators were outlined as baselines from the discovered process models.
  - *Radiology* and *Consults* process maps had fewer case variants than other process maps generated for *Lab Services* and *RxOut*.
  - We found that cases that keep looping usually are discontinued and consequently, destined to fail. These are examples of possible anomalies that could become HIT hazards.
  - The frequencies are higher on those activities related to the *CPRSOrder* domain. Therefore, one should give special attention to the IT resources that maintain data in this domain including storage, network bandwidth, memory and processor speed.
- **Combining event sequences with a powerful visualization tool like process mining can reveal important aspects of the data that are hard to interpret otherwise!**

# Acknowledgments

This work was performed at  
**Oak Ridge National Laboratory**  
and was sponsored by the  
**Veterans Affairs Administration**





## **Feature Engineering and Process Mining to Enable Hazard Detection in Health Information Technology**

**Ozgur Ozmen, MS, MISE, PhD<sup>1</sup>, Hilda B. Klasky, MS<sup>1</sup>, Olufemi A. Omitaomu, PhD<sup>1</sup>, Mohammed Olama, PhD<sup>1</sup>, Teja Kuruganti, PhD<sup>1</sup>, Laura L. Pullum, MSOR, MBA, MSGeo, DSc<sup>1</sup>, Merry Ward, MS, PhD<sup>2</sup>, Jean M. Scott, MS<sup>2</sup>, Angela Laurio, RN, DrPH<sup>2</sup>, Frank Drews, PhD<sup>2</sup>, Jonathan R. Nebeker, MS, MD<sup>2</sup>**

**<sup>1</sup>Oak Ridge National Laboratory, Oak Ridge, TN, USA; <sup>2</sup>US Department of Veterans Affairs, Washington, DC, USA**

### **Abstract**

*In this work, we aim to enhance the reliability of health information technology (HIT) systems by detection of plausible HIT hazards in clinical order transactions. In the absence of well-defined event logs in corporate data warehouses, our proposed approach identifies relevant timestamped data fields that could indicate transactions in the clinical order life cycle generating raw event sequences. Subsequently, we adopt state transitions of the OASIS Human Task standard to map the raw event sequences and simplify the complex process that clinical radiology orders go through. We describe how the current approach provides the potential to investigate areas of improvement and potential hazards in HIT systems using process mining. The discussion concludes with a use case and opportunities for future applications.*



## Process mining in Healthcare – A Case Study for the Corporate Data Warehouse of the Veterans Affairs Office.



Hilda B. Klasky  
Ozgur Ozmen

Approved for public release.  
Distribution is unlimited.

September 2019

# U.S. Department of Energy

## Office of Scientific and Technical Information

osti.gov

**We can discuss further in the Process  
Mining Camp community / Slack**

**You can also email me:  
[klaskyhb@ornl.gov](mailto:klaskyhb@ornl.gov)**

**Thank you!**

Questions?