

Referrals & Orders: How to ask for stuff in FHIR

Lloyd McKenzie – Dogwood Health Consulting




HL7 FHIR DevDays 2023 | Hybrid Edition, Amsterdam | June 6–9, 2023 | @HL7 | @FirelyTeam | #fhirdevdays | www.devdays.com

ORGANIZED BY

firely

HL7[®]
International

Who am I?

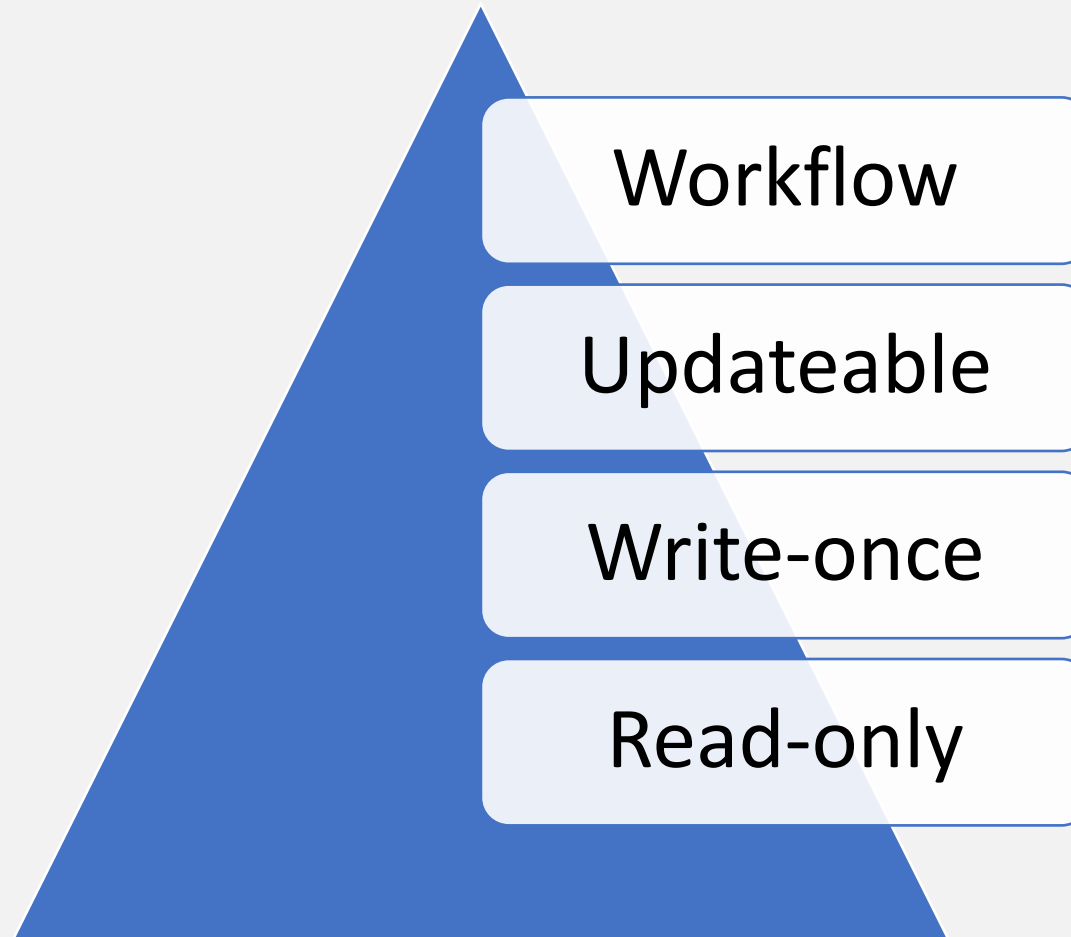
- Name: **Lloyd McKenzie**
- Role: Chief Standards Officer
- Company:  **DOGWOOD**
Health Consulting
- Background:
 - One of FHIR's initial editors
 - Cochair of FMG and FHIR Infrastructure
 - Heavily involved in healthcare interoperability for over 20 years
 - Unofficial FHIR community manager
 - **Lead on FHIR Workflow project**



Learning Objectives

- Explain difference between ‘Ordering’ and ‘Seeking fulfillment’
- List some of the mechanisms for asking for something to be done
- Understand the pros and cons between those mechanisms
- Describe how Task interconnects ‘placer’ and ‘filler’
- Differentiate between types of definition, request, and event relationships
- Know where to look for more guidance

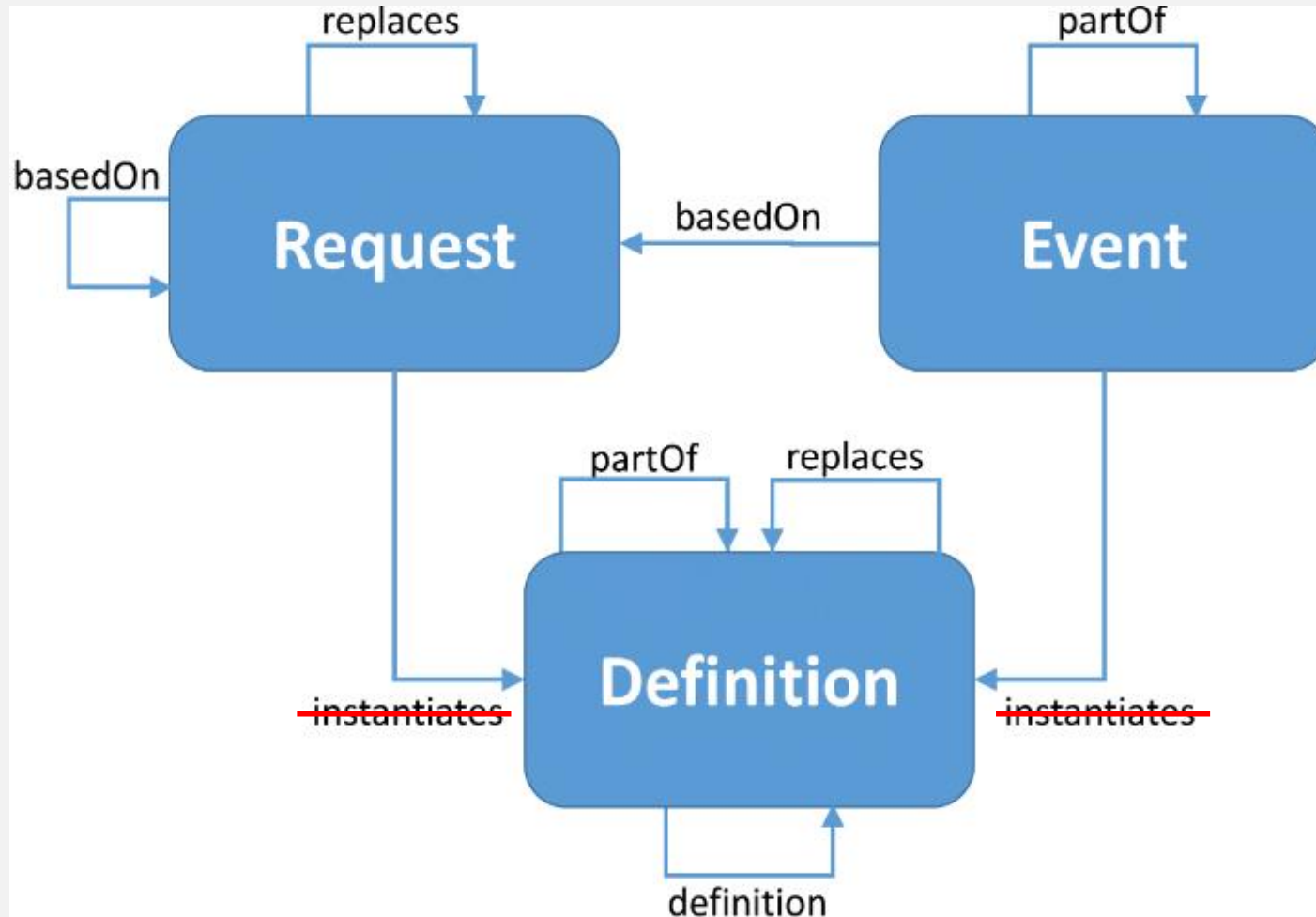
FHIR EHR maturity



Asking for stuff

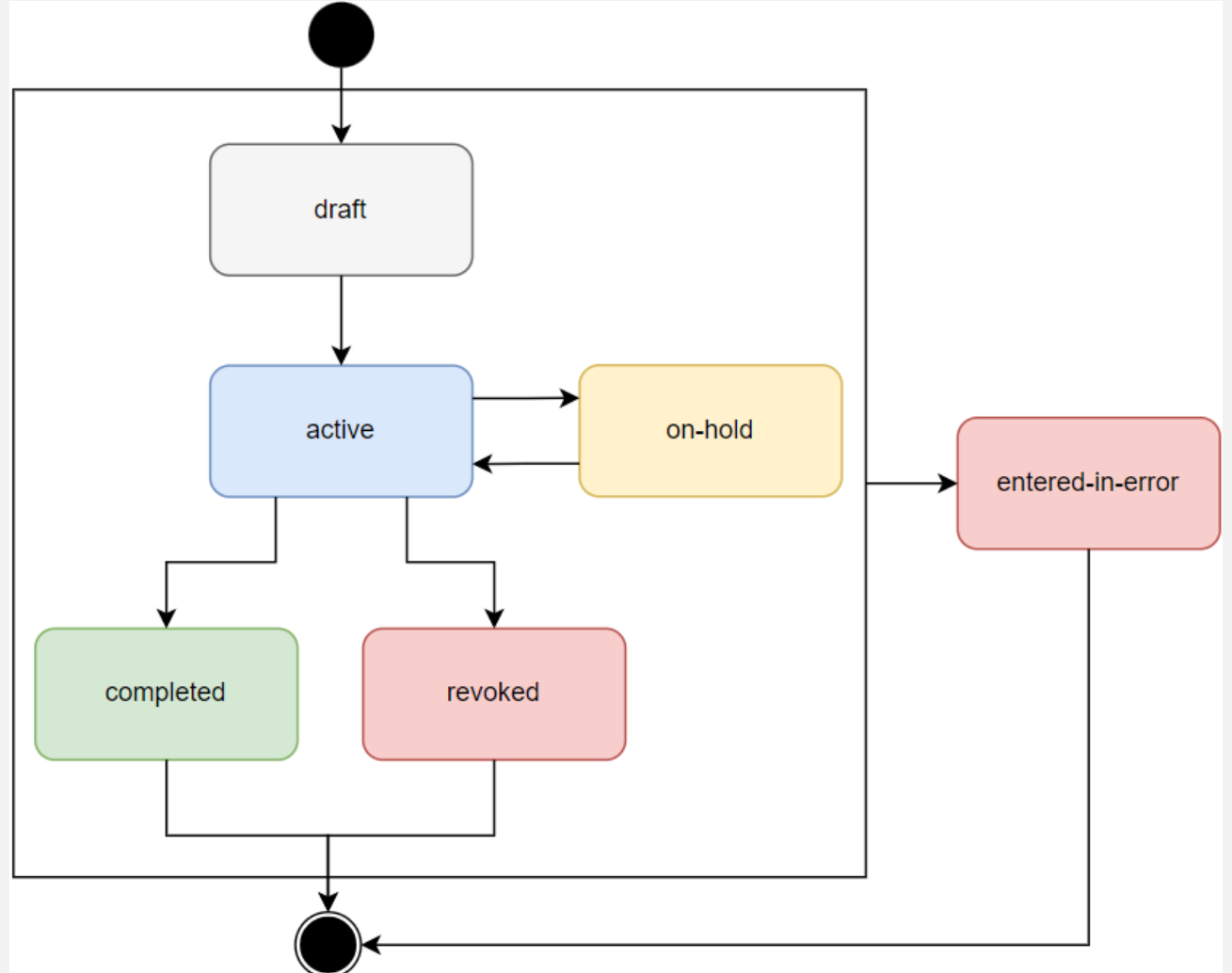
- Here's a drug prescription/lab order
- Referrals
- “Please fill out this form”
- Please suspend this order
- Please discharge this patient
- Please call me

Workflow resources



Request pattern

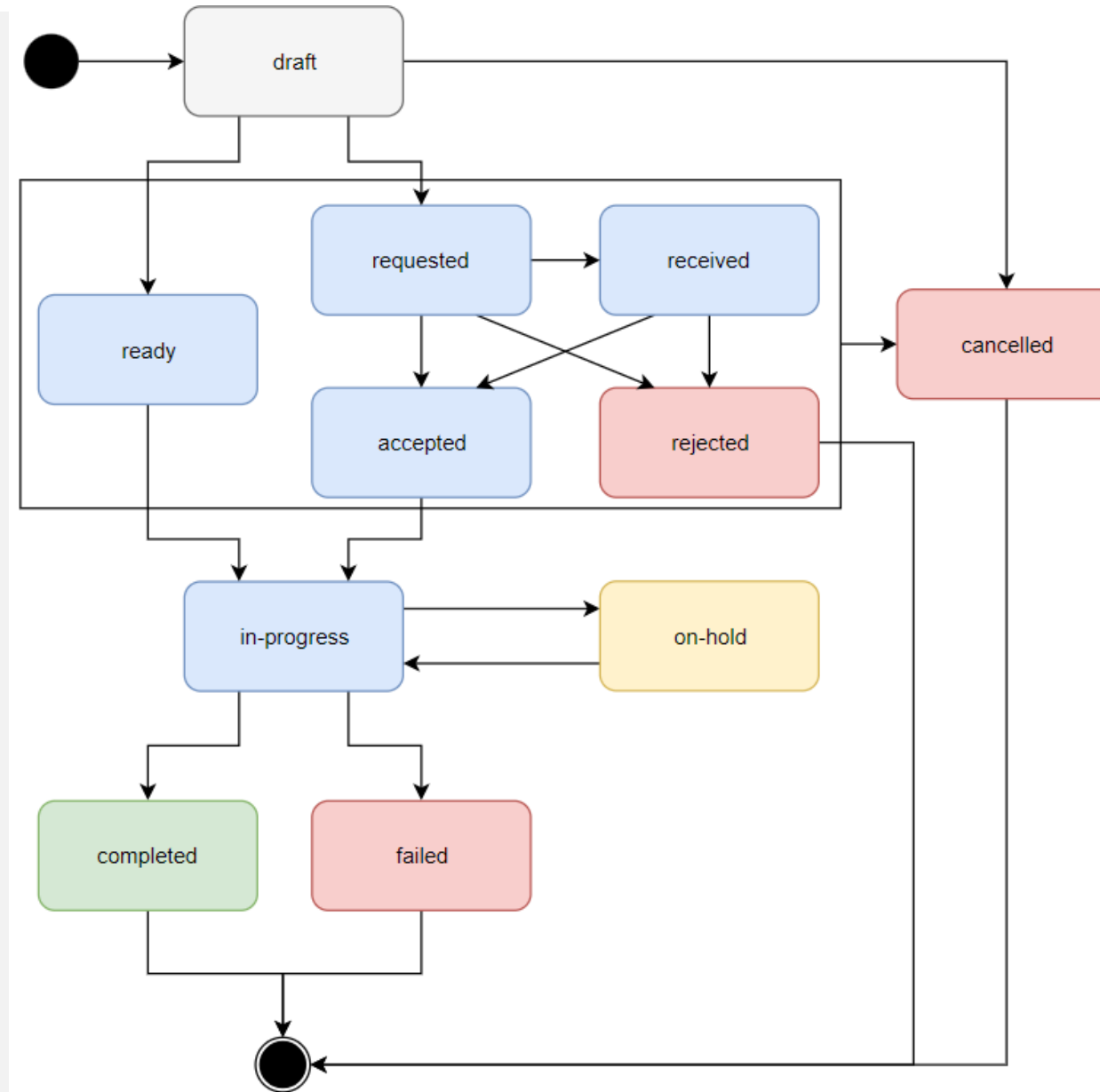
Request (Base) «Pattern»
identifier : Identifier [0..*] basedOn : Reference [0..*] « Request » replaces : Reference [0..*] « Request » groupIdIdentifier : Identifier [0..1] status : code [1..1] « RequestStatus! » statusReason : CodeableConcept [0..1] intent : code [1..1] « RequestIntent! » priority : code [0..1] « RequestPriority! » doNotPerform : boolean [0..1] category : CodeableConcept [0..*] code : CodeableConcept [0..1] product : CodeableReference [0..1] « BiologicallyDerivedProduct Device DeviceDefinition Medication NutritionProduct Substance » subject : Reference [1..1] « Patient Group » encounter : Reference [0..1] « Encounter » occurrence[x] : DataType [0..1] « dateTime Period Timing » authoredOn : dateTime [0..1] requester : Reference [0..1] « Practitioner PractitionerRole Organization Patient RelatedPerson Device » reported[x] : DataType [0..1] « boolean Reference(Patient RelatedPerson Practitioner PractitionerRole Organization) » performerType : CodeableConcept [0..1] performer : Reference [0..1] « Practitioner PractitionerRole Organization CareTeam HealthcareService Patient Device RelatedPerson » deliverTo : Reference [0..*] « Patient RelatedPerson Practitioner PractitionerRole Organization Location HealthcareService CareTeam Device Endpoint » reason : CodeableReference [0..*] « Condition Observation DiagnosticReport DocumentReference » insurance : Reference [0..*] « Coverage ClaimResponse » supportingInfo : Reference [0..*] « Any » note : Annotation [0..*] relevantHistory : Reference [0..*] « Provenance »



But...

- Where are:
 - 'Accepted',
 - 'In progress',
 - ...?

• Task

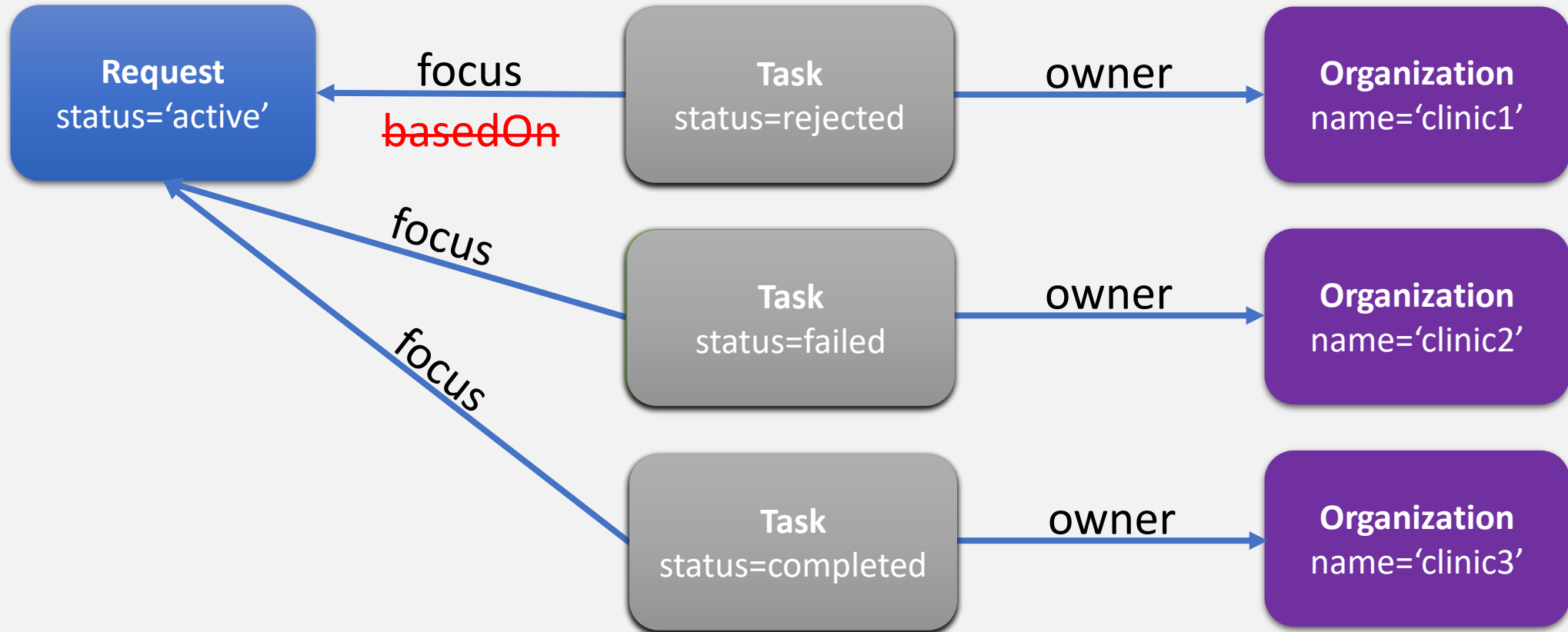


Why Task?

- What if performer is selected after order signed?
- What if initial recipient says 'no'?
- What if you don't want the whole order completed?
- What if there's extra information performer needs?
 - (not identified at order sign)
- What if there's communication that needs to happen during execution?

- Task is only resource to be Request and Event

Seeking Fulfillment



Task

Task (DomainResource)

```

identifier : Identifier [0..*]
instantiatesCanonical : canonical [0..1] « ActivityDefinition »
instantiatesUri : uri [0..1]
basedOn : Reference [0..*] « Any »
groupIdIdentifier : Identifier [0..1]
partOf : Reference [0..*] « Task »
status : code [1..1] « TaskStatus! »
statusReason : CodeableReference [0..1] « ; TaskStatusReason?? »
businessStatus : CodeableConcept [0..1] « TaskBusinessStatus?? »
intent : code [1..1] « TaskIntent! »
priority : code [0..1] « RequestPriority! »
doNotPerform : boolean [0..1]
code : CodeableConcept [0..1] « TaskCode?? » « C »
description : string [0..1]
focus : Reference [0..1] « Any » « C »
for : Reference [0..1] « Any »
encounter : Reference [0..1] « Encounter »
requestedPeriod : Period [0..1]
executionPeriod : Period [0..1]
authoredOn : dateTime [0..1] « C »
lastModified : dateTime [0..1] « C »
requester : Reference [0..1] « Device | Organization | Patient |
  Practitioner | PractitionerRole | RelatedPerson »
requestedPerformer : CodeableReference [0..*] « Practitioner |
  PractitionerRole | Organization | CareTeam | HealthcareService |
  Patient | Device | RelatedPerson; ProcedurePerformerRoleCodes? »
owner : Reference [0..1] « Practitioner | PractitionerRole |
  Organization | CareTeam | Patient | RelatedPerson »
location : Reference [0..1] « Location »
reason : CodeableReference [0..*] « ; TaskReason?? »
insurance : Reference [0..*] « Coverage | ClaimResponse »
note : Annotation [0..*]
relevantHistory : Reference [0..*] « Provenance »
  
```

Restriction

```

repetitions : positiveInt [0..1]
period : Period [0..1]
recipient : Reference [0..*] « Patient | Practitioner | PractitionerRole |
  RelatedPerson | Group | Organization »
  
```

Performer

```

function : CodeableConcept [0..1] « TaskPerformerFunctionCode?? »
actor : Reference [1..1] « Practitioner | PractitionerRole |
  Organization | CareTeam | Patient | RelatedPerson »
  
```

Input

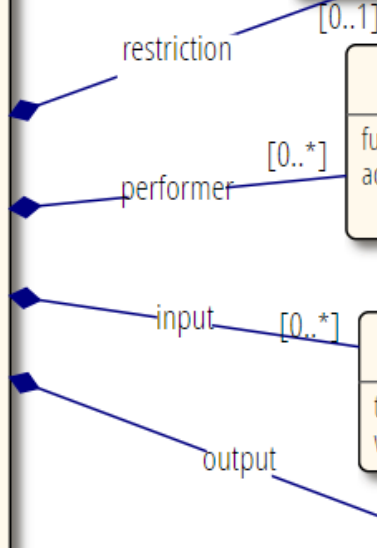
```

type : CodeableConcept [1..1] « TaskInputParameterType?? »
value[x] : * [1..1]
  
```

Output

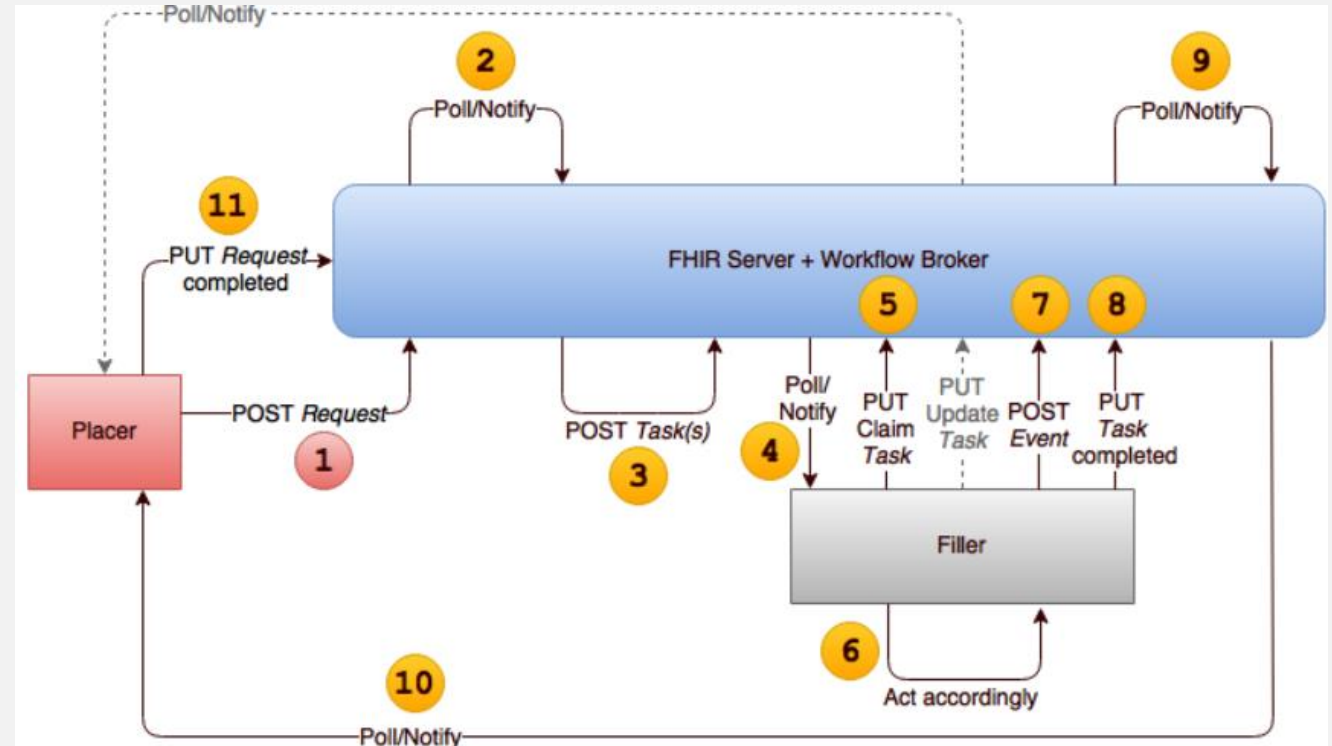
```

type : CodeableConcept [1..1] « TaskOutputParameterType?? »
value[x] : * [1..1]
  
```



Where does the Task live?

- Filler (Placer monitors)
- Placer (Filler monitors)
- Intermediary (Both monitor)



Do you have to use Task?

- Can just post the ServiceRequest to filler
 - Must have 'actionable' tag
 - Can also post to placer with assigned performer (polling/subscription)
- Use messaging
- Use a custom operation?
- Use a document?

Do you have to have a request?

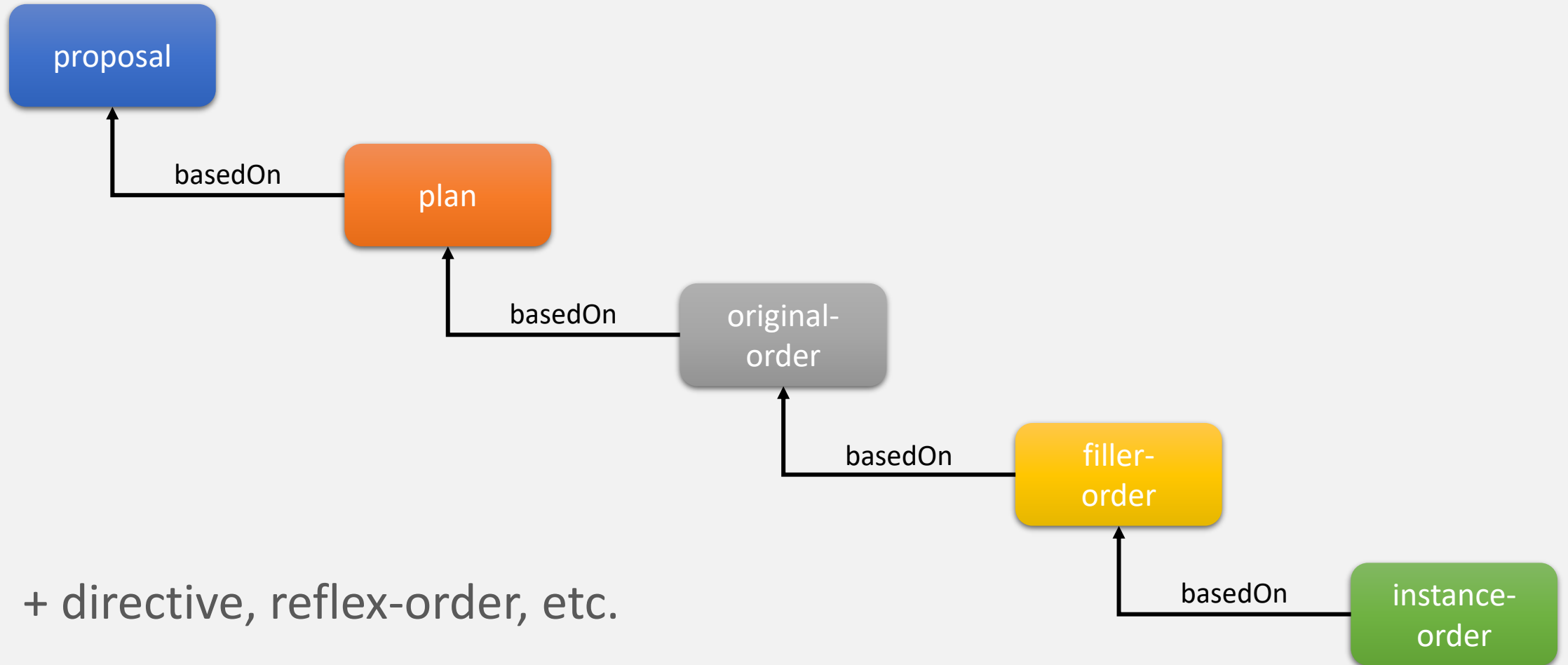
NO:

- Request a status change
- Simple request that doesn't require 'authorization'
 - Change this bed
 - Fill out this form

But I need to send more than just the Request...

- Request.supportingInfo
- Task.input
- Query what you need
- Composition to tell a story

Intent



What replaces instantiates?

- Workflow extensions
 - **adheres-to** – event does what definition says
 - **complies-with** – Request asks for what Definition says
 - **generated-from** – Request was created from Definition
 - **shall-comply-with** – When performing Request, follow Definition
 - **triggered-by** – Event or Request came into being because of Definition

There are FHIR pages for this...

- <https://build.fhir.org/workflow-communications.html>
 - <https://build.fhir.org/workflow-ad-hoc.html>
 - <https://build.fhir.org/workflow-management.html>
- Also: <https://chat.fhir.org/#narrow/stream/179196-workflow>
- Or come join our calls
 - Mondays, 2 Eastern
 - Details on FHIR-I Confluence page - <https://confluence.hl7.org/display/FHIRI>

What did you learn?

- Order = authorization; Fulfillment = workflow process
- Can seek fulfillment by
 - REST (variety of architectures), messaging, operation
- FHIR offers more flexibility than “referral package”
- Task is the one resource intended to be updated by ‘placer’ and ‘filler’
- Workflow can tie resources together in complex chains
- For guidance, look to Workflow in spec and chat.fhir.org

Q&A

- lloyd@dogwoodhealthconsulting.com
- Or better yet, include the community and ask/discuss on chat.fhir.org



ORGANIZED BY

