

SinusitisTherapy

DTRules rules reference — data model & decision tables

How this rule set is built

A DTRules application is two artifacts authored in spreadsheets: an **Entity Data Dictionary** that defines the data, and a set of **decision tables** that define the logic. This document lists both, plus how they drive an interactive interview.

The Entity Data Dictionary (EDD) — what it contains

The EDD is the data model. It declares the **entities** (things the rules reason about, e.g. a patient) and, for each, its **attributes**: a name, a type (string, integer, decimal, boolean, date, an entity reference, or an array), an optional default, and whether the field is an *input* the rules read or an *output* they produce. The decision tables may only reference fields declared here — the EDD is the shared vocabulary.

An attribute can also be marked **collected**: rather than arriving as bulk input, the engine asks the user for it. Collected fields carry question metadata — a prompt, a type (`multiple_choice`, `ascii`, `number`, `date`), choice options, and for numbers a lab-style normal range.

Decision tables — what they contain

Each decision table is a grid. The top rows are **conditions** (when something is true) and the bottom rows are **actions** (what to do); every numbered **column is one rule**. A condition cell is **Y** (must be true), **N** (must be false), or **–** (don't care). An action cell is marked when that action fires for the column. Each condition and action is written in DTRules' Expression Language (EL), shown in the grid.

- **Policy** — **FIRST** fires the first matching column, **ALL** fires every matching column in order, **BALANCED** expects exactly one column to match.
- **Initial actions** run once before the columns; **contexts** push an entity (e.g. "for all active medications") so the table evaluates against each.
- Tables call other tables with `perform`, composing larger logic from small, readable pieces.

The interview UI is automatic

There is no hand-built form. When the engine *executes the decision tables* and a rule reads a collected field that hasn't been supplied, it pauses and asks that field's question — on the command line or in the browser — then resumes. The screens a user sees are a by-product of which fields the tables actually reach: change the tables or the EDD questions and the interview changes with them, with no UI code to update.

Entity Data Dictionary

The data model the decision tables read and write. ● marks a field the interview collects from the user.

constants

Field	Type	Default	I/O	Asked as	Description
adult_age	integer	18	input	—	Age at which a patient is treated as an adult
dose_high_age	integer	60	input	—	Upper bound of the standard-dose age band
dose_low_age	integer	15	input	—	Lower bound of the standard-dose age band
dosing_interval_hours	integer	24	input	—	Standard dosing interval in hours
reduced_dose	integer	200	input	—	Reduced dose in mg
renal_ccr_threshold	double	50.0	input	—	CCr below which renal adjustment applies
renal_pcr_threshold	double	1.4	input	—	PCr above which renal adjustment is considered
standard_dose	integer	500	input	—	Standard dose in mg
therapy_duration_days	integer	14	input	—	Standard therapy duration in days

medication

Field	Type	Default	I/O	Asked as	Description
name	string	—	in-out	—	Drug name (brand or generic)

patient

Field	Type	Default	I/O	Asked as	Description
active_medications	array of medication	—	in-out	—	Medications the patient is currently taking
● age	integer	0	in-out	number in years	Patient age in years
● diagnosis	string	Acute Sinusitis	in-out	ascii	Working diagnosis, e.g. Acute Sinusitis
● lean_body_weight	double	0.0	in-out	number in kg	Lean body weight in kg (used by the CCr formula)
● pcr	double	0.0	in-out	number normal 0.7–1.3 mg/dL	Plasma creatinine level (mg/dL)
● penicillin_allergic	boolean	false	in-out	multiple_choice (Yes, No)	True if the patient is penicillin-allergic

result

Field	Type	Default	I/O	Asked as	Description
ccr	double	0.0	in-out	—	Service 2 output: creatinine clearance (mL/min)
dose_mg	integer	0	output	—	Service 1 output: single dose in mg
duration_days	integer	0	output	—	Service 1 output: therapy duration in days
frequency_hours	integer	0	output	—	Service 1 output: hours between doses
rationale	array of string	—	in-out	—	Human-readable rationale for each service decision (explainability)
recommended_drug	string	—	in-out	—	Service 1 output: drug of choice
renal_adjustment	boolean	false	output	—	True if the dose was reduced for renal impairment
warnings	array of string	—	in-out	—	Service 3 output: drug-interaction warnings

Decision Tables

6 tables. Conditions (top) decide which rule columns apply; actions (bottom) run for the marked columns.

Y condition must be true **N** must be false – don't care ✓ action fires

1000 · Determine_Therapy

FIRST

#	Condition	1
1	patient.diagnosis is equal to ignore case "Acute Sinusitis" Always run the full therapy workup	Y
ACTION		1
1	perform Determine_Creatinine_Clearance Service 2 first: CCr feeds the dosing rules	✓
2	perform Determine_Medication_And_Dosing Service 1: drug of choice and dosing	✓
3	perform Check_Drug_Interactions Service 3: drug-interaction warnings	✓

2000 · Determine_Creatinine_Clearance

FIRST

#	Condition	1
1	true Always compute CCr	Y
ACTION		1
1	set result.ccr = ((140 - patient.age) * patient.lean_body_weight) / (patient.pcr * 72) Cockcroft-Gault style: CCr = ((140 - age) * lean body weight) / (PCr * 72)	✓
2	add "CCr estimated via Cockcroft-Gault from age, lean body weight, and plasma creatinine." to result.rationale Rationale: CCr method	✓

3000 · Determine_Medication_And_Dosing

FIRST

#	Condition	1
1	true Always run	Y
ACTION		1
1	perform Select_Medication Pick the drug of choice	✓
2	perform Determine_Dose Pick the dose / frequency / duration	✓

3010 · Select_Medication

FIRST

#	Condition	1	2	3
1	<code>patient.penicillin_allergic is true</code> Penicillin allergy overrides age (beta-lactam cross-reactivity)	Y	–	–
2	<code>patient.age >= constants.adult_age</code> Adult (>= 18)	–	Y	N
ACTION		1	2	3
1	<code>set result.recommended_drug = "Levofloxacin"</code> Fluoroquinolone for penicillin-allergic patients	✓		
2	<code>set result.recommended_drug = "Amoxicillin"</code> Adult first-line		✓	
3	<code>set result.recommended_drug = "Cefuroxime"</code> Pediatric first-line			✓
4	<code>add "Levofloxacin selected: fluoroquinolone avoids beta-lactam cross-reactivity in a penicillin-allergic patient." to result.rationale</code> Rationale: penicillin-allergic	✓		
5	<code>add "Amoxicillin selected: first-line beta-lactam for a non-allergic adult." to result.rationale</code> Rationale: adult first-line		✓	
6	<code>add "Cefuroxime selected: first-line for a non-allergic pediatric patient." to result.rationale</code> Rationale: pediatric first-line			✓

3020 · Determine_Dose

FIRST

#	Condition	1	2	3
1	<code>patient.pcr > constants.renal_pcr_threshold and result.ccr < constants.renal_ccr_threshold</code> Renal impairment: PCr > 1.4 and CCr < 50	Y	–	–
2	<code>patient.age >= constants.dose_low_age and patient.age <= constants.dose_high_age</code> Standard-dose age band (15..60)	–	Y	N
ACTION		1	2	3
1	<code>set result.dose_mg = constants.reduced_dose</code> Renal-reduced dose	✓		
2	<code>set result.dose_mg = constants.standard_dose</code> Standard dose for 15..60		✓	
3	<code>set result.dose_mg = constants.reduced_dose</code> Reduced dose for <15 or >60			✓
4	<code>set result.renal_adjustment = true</code> Flag the renal adjustment	✓		
5	<code>set result.frequency_hours = constants.dosing_interval_hours</code> Dosing interval (hours)	✓	✓	✓
6	<code>set result.duration_days = constants.therapy_duration_days</code> Therapy duration (days)	✓	✓	✓

4000 · Check_Drug_Interactions

FIRST

contexts:
for all patient.active_medications

#	Condition	1	2	3	4	5	6
1	result.recommended_drug is equal to ignore case "Levofloxacin" Recommended drug is Levofloxacin	Y	Y	Y	–	–	–
2	result.recommended_drug is equal to ignore case "Amoxicillin" Recommended drug is Amoxicillin	–	–	–	Y	Y	–
3	result.recommended_drug is equal to ignore case "Cefuroxime" Recommended drug is Cefuroxime	–	–	–	–	–	Y
4	medication.name is equal to ignore case "Coumadin" or medication.name is equal to ignore case "Warfarin" Active med is an anticoagulant	Y	–	–	Y	–	–
5	medication.name is equal to ignore case "Theophylline" Active med is theophylline	–	Y	–	–	–	–
6	medication.name is equal to ignore case "NSAID" or medication.name is equal to ignore case "Ibuprofen" or medication.name is equal to ignore case "Naproxen" Active med is an NSAID	–	–	Y	–	–	–
7	medication.name is equal to ignore case "Methotrexate" Active med is methotrexate	–	–	–	–	Y	–
8	medication.name is equal to ignore case "Probenecid" Active med is probenecid	–	–	–	–	–	Y
ACTION		1	2	3	4	5	6
1	add "Levofloxacin may potentiate the anticoagulant effect of warfarin/Coumadin, raising INR and bleeding risk. Monitor coagulation closely." to result.warnings Levofloxacin + anticoagulant	✓					
2	add "Levofloxacin may increase theophylline plasma levels; monitor for theophylline toxicity." to result.warnings Levofloxacin + theophylline		✓				
3	add "Concomitant NSAIDs with Levofloxacin may increase the risk of CNS stimulation and seizures." to result.warnings Levofloxacin + NSAID			✓			
4	add "Amoxicillin may enhance the anticoagulant effect of warfarin/Coumadin; monitor INR." to result.warnings Amoxicillin + anticoagulant				✓		
5	add "Amoxicillin may reduce renal clearance of methotrexate, increasing methotrexate toxicity." to result.warnings Amoxicillin + methotrexate					✓	

#	Condition	1	2	3	4	5	6
6	add "Probenecid reduces renal excretion of Cefuroxime, raising its plasma concentration." to result.warnings Cefuroxime + probenecid						✓