

# **An illustration of an ISO/IEC 15944-21 Open-edi\* Distributed Business Transaction Repository (OeDBTR) immutable record of business transactions in accounting collaboration space**

**April 2021**

A research collaboration  
led by Jonas Sveistrup Soegaard and William McCarthy  
with Lasse Herskind and G. Ken Holman

<https://gkenholman.page.link/oedbtr-202104-pdf> - the example scenario PDF file

<https://gkenholman.page.link/oedbtr-202104-video> - the video of this transcript

<https://gkenholman.page.link/oedbtr-202104-git> - the publishing repository for the project

\* ISO/IEC 15944-4 Open-edi Business Transaction Ontology (OeBTO)

ISO/IEC 14662 Open-edi Reference Model

<https://standards.iso.org/ittf/PubliclyAvailableStandards>

## **Narration Video Transcript**

### **00:00 Cover**

This is a narration of the tracking through example business transactions in accounting collaboration space. This demonstrates the multiple state machines used in an Open-edi Distributed Business Transaction Repository (OeDBTR) as defined in the ISO/IEC 15944 Part 21 international standard. These state machines change through a series of economic resources, events, and agents as defined by the Open-edi Business Transaction Ontology (OeBTO) as specified in ISO/IEC 15944 Part 4. The entire 15944 family of standards is based on the principles of the Open-edi Reference Model specified in ISO/IEC 14662. These standards documents are available free of charge, though Part 21 still sits in a pre-publishing stage at the time of this recording.

### **01:04 Inside Cover**

An Open-edi Distributed Business Transaction Repository records in an immutable store the sequence of business events making up a business transaction. An independent observer can look back at any of the recorded events to establish the state of all of the economic

resources, events, and agents involved in a business transaction. This embedded diagram describes the ISO-standard Open-edi Business Transaction Ontology. Each of these boxes is a primitive in the ontology, and the lines between them are the relationships between the primitives. Each primitive has its own definition of a state machine and one or more states that may change, and zero or more properties that can have values that may change.

Each business event triggers one or more stimuli accompanied with a set of property values. Each stimulus impacts one or more of the state machines. In this demonstration the immutable store of these state machines is a PDF file. The status of all of the state machines at any given point in the history of economic exchange activity can be queried in the PDF simply by going back to the page reporting the end of the state changes.

The first business event is at the back of this stack that is shown, and the most recent event is at the front. Zooming in on the primitives of the ontology we see Persons (or economic agents) in different economic roles engage in economic events of different economic event types in fulfilment of economic commitments as part of an economic contract. These economic events effect the exchange of economic resources of different economic resource types. Note that not all of these primitives need to be reified in state machines for the business transaction to be complete and useful.

To simulate this time sequence record, the PDF file I will review has one page per business event and that page contains the current state and set of properties of all of the instantiated state machines in support of the primitives of the business transaction. Each instantiation of a primitive is shown as a box. The changes of state and property values are highlighted in red. The first page of the PDF has a legend and a definition of all of the available state machines and their transitions based on possible stimuli. Each business event generates one or more stimuli and all of the machines are checked for a response to each stimulus. The subsequent pages of the PDF represent the business transaction's business events from the back to the front in a time sequence. From the first page to the next, and the next, and so on, back to front.

The eye here represents the independent view of the business transaction, which could be either buyer or seller or auditor or any other party interested in the state of the state machine of each of the business event primitives of the business transaction. This independent view can interpret the single representation of all states at a point in time however applicable to the viewer's role, if any, in the transaction.

Consider this metaphor below of having two different prisms each projecting a differing perspective from a single source of truth that all trading partners and other interested parties access. No single perspective is in any way "better" or "preferred" over the other perspectives. In this demonstration, this particular OeDBTR records a requited exchange of cash and goods from an independent perspective. A buyer sees that requited exchange of cash and goods through their prism as a purchase outlay of cash, while the seller sees the very same requited exchange of cash and goods through their prism as a sale inflow of cash. But these are just projections of the independent requited exchange in the OeDBTR.

So now let us look at the individual state machines of the components of the business transaction in this example.

## **06:26 Legend and state machine documentation**

Opening the PDF file brings us to the legend and state machine documentation on the first page.

On each page the top line titles the particular OeDBTR set of state machine definitions that is in play for the choreography of business events, and the title of the particular scenario of business events that is being played out in the pages.

At the top is the legend with a prototypical state machine, and below that, all of this DBTR's ten state machine definitions. Each documents the available states and how any given stimulus sent to the state machine changes the current state to a new state.

Let's look in closer detail.

In the top right, a circled letter R indicates that the state machine is reusable, that is, an instance of the state machine can be reused across multiple scenarios once it has been defined. In the top left, a circled letter C indicates that the state machine is completed, that is, a particular instance of the state machine will never be able to change its state or its properties as it has reached an end state. The identifier for a particular instance of a state machine is underscored and boldfaced.

Visual cues indicate changes in values. When viewed in colour, any change is marked in red. When viewed in black and white, state changes are marked with a checked box and a dashed box around the state name, and property changes are marked with a tilde character. Old state changes are marked with an X, the current state when it does not change is boxed in solid, and old property values that have not changed are marked with an equal sign.

Looking at the definitions of each of the state machines, we see the name of the ontology primitive and all of that state machine's states. Each state has listed below it the transition based on a given stimulus and what the new state is to be. So if the state of the economic commitment machine is "proposed" and the stimulus that arrives is "accept commitment", "proposed" plus "accept commitment" equals "specified" as the new state for this state machine.

On the remaining pages, the choreography of business events is the conversation for action for the given scenario. In this scenario, the wholesale sale of bicycles to a bicycle retail outlet, most business events involve the exchange of business documents between the buyer, the bike shop, and the seller, the bicycle manufacturer.

But this conversation for action begins with a few "behind the scenes" business events, not associated with the exchange of business documents. These are used to establish the reusable state machine instances that are needed later.

## **10:08 Event #1 - preparation events**

Each statement in the conversation for action is shown next to the business event sequence number. Listed below the statement are all of the stimuli triggered by this business event.

First, we need a resource type with which to value other resource types. The Dai resource type is instantiated as being a cryptocurrency that is par with the US dollar.

Note the convention of changed state machines in dashed red and uninstantiated state machines in gray for those primitives that are not yet in play.

## **10:55 Event #2**

Next, we establish all of the economic agent types. We end up with three instances of agent types, one with the role of being the buyer, one with the role of being the seller, and one being the role of the tax authority. These machines are changed and so they are boxed in red dashes, while the unchanged state machine in play is boxed in black solid.

## **11:30 Event #3**

Next, we establish all of the economic agents that are using those economic agent types. Note the naming convention of 3-digits distinguishing the parties. In this demonstration we have adopted the 3-digit suffix in identifiers for continuants, that is, for things that retain their identity even though its states change. In these cases, the continuant is referenced by its identity instead of by its name. Party100 is "Bill's Bikes Manufacturing" in the role of the seller, which is one of the previously-established economic agent types. Party200 is "Jonas's Bike Shop" in the role of the buyer, and his wallet is initialized with 100000 Dai to start. Party 300 is the Danish Tax Agency in the role of the tax authority. The balances of both the seller and the tax agency start at zero for illustration purposes.

## **12:52 Event #4**

Next, we establish all of the product economic resource types. GreenSportBike and RedSportBike each have a suggested price and a suggested tax in the given currency. Green sport bikes have a suggested retail price of 1200 Dai, while red sport bikes have suggested price of 800 Dai.

## **13:23 Event #5**

Next we establish an inventory of 11 available green sport bikes, each with a unique serial number that in this example also serves as its identifier. All are owned by Party100, the seller party.

## **13:44 Event #6**

Next we establish an inventory of red sport bikes in the same fashion.

## **13:52 Event #7**

Finally in our prelude we establish two economic event types that will be needed in the conversation for action: a delivery event for wholesale deliveries, and a payment event type where the payment is in Dai.

## **14:13 Event #8 - Catalogue Request**

The conversation for action begins with the buyer asking "What inventory do you have?" by sending a "Catalogue Request" to the seller that triggers a "request a catalogue" stimulus. As is true for all of the business events in this conversation, an example document model for each event is available by clicking on the link in the PDF file. For this first conversation event such a document is the ISO standardized Universal Business Language (UBL) Version 2.1 Catalogue Request. Note that the referenced file is very large and may take some moments to load. This report enumerates all of the semantic components of UBL business documents. In this case for the Catalogue Request.

This stimulus, request a catalogue, triggers the process overview and process detail for the conversation identified as "Process1". In overview, the planning state has begun, and in detail, the catalogue has been requested. None of the other state machines has changed because requesting a catalogue has no impact on any of them.

Note the convention in this demonstration of the using a single digit suffix for occurrents, that is, for things that either exist only for a period of time and then disappear when no longer needed, or for things that are fungible and have no unique identity. Occurrents differ from continuants in that the continuants persist, have unique identity, and are not fungible.

## **16:21 Event #9 - Catalogue**

The seller responds "I have red and green bikes" by sending the document "Catalogue" to the buyer. Here are the semantic components of a UBL catalogue. In overview there is no change as it remains in the planning state. In detail the process has changed state to the catalogue having been received. Again, none of the other state machines has changed.

## **17:09 Event #10 - Request for Quotation**

So the buyer likes what he sees in the catalogue and so he asks "Do you have the green bikes that I want?" by establishing an economic commitment as part of an economic contract. The commitment is planning to deliver on June 15 ten green sport bikes by sending a "Request for Quotation" document to the seller. The process overview moves from "planning" to "identification" and the process detail moves from "catalogue received" to "quotation requested". At this point the economic contract dated on June 1 is instantiated making reference to the planned delivery "PlanDel1".

## **18:10 Event #11 - Quotation**

The seller looks at his inventory and responds "Yes, I have the bikes that you want" by returning the "Quotation" document to the buyer. This event has two stimuli. One stimulus establishes the value of the contract to be 12000 Dai and the associated 3000 Dai in tax. The other stimulus moves the process overview from "identification" to "negotiation" and moves the process detail from "quotation requested" to "quotation received".

### **18:51 Event #12 - Order**

The buyer responds to this with "Okay, here is an order for green bikes" by returning the document "Order" to the seller. Four stimuli are sent to the state machines. The contract is modified with a new economic commitment named "PlanPay1" of a new instantiated planned payment for June 10 of the required amounts using a new economic resource of an allocation of sufficient funds to cover the purchase. The bikes cost 12000 Dai and the tax is 3000 Dai and so the allocation amount is 15000 Dai. This allocation is from the buyer party Party200 and is initialized as being available for the transaction. The process overview remains in the "negotiation" state and the process detail moves to the "order-received" state.

### **20:03 Event #13 - Order Response Simple**

The seller responds "Deal! The order is acceptable" by using an Order Response Simple document. In so doing the planned delivery and planned payment proposed in the contract each become "specified" instead of "proposed". At the same time, the economic resources of the individual bicycles are now "reserved" by the seller, instead of just "available", so as not to be available for any other party. The actual price and the actual tax are recorded in case they might be different from the suggested price and tax. One of the stimuli moves the process overview to the "actualization" state and the process detail to "order accepted" state.

### **21:01 Event #14 - Remittance Advice**

The buyer responds to the order "Okay, here is the payment" using the "Remittance Advice" document and the prepaid amount of cryptocurrency required. This advance payment is indicated in a new economic event of type "PaymentType" named "Pay1" as a fulfillment of the planned payment "PlanPay1". This points to the stock flow allocation of funds named "Dai1" that now is marked as being reserved so as not to be available for any other party. The funds in that allocation, the 15000 Dai, are removed from the buyer's wallet and sit now in the contract's wallet as a kind of blockchain-based escrow out of the hands of the buyer and the seller. The process overview remains in the "actualization" state and the process detail moves to the "paid" state.

### **22:24 Event #15 - Despatch Advice**

The seller comes back to the buyer "Thank you, I have despatched 10 green bikes" with a

"Despatch Advice" document. The new economic event of type "DeliveryType" named "Del1" is instantiated as a fulfillment of the planned delivery "PlayDel1". The process overview remains in the "actualization" state and the process detail moves to the "despatched" state.

### **22:59 Event #16 - Receipt Advice**

The buyer comes back to the seller "Thank you, I have received 10 green bikes" with a "Receipt Advice" document. This triggers a number of changes. The process overview remains in the "actualization" state and the process detail moves to the "goods received" state. The delivery economic event moves to "event complete", which also completes the planned delivery economic commitment. Importantly, all of the economic resources of the bikes themselves move to the "exchanged" final state with the owner changing to the buyer party, the actual price at which the items were sold recorded, and the indication of which stock flow delivery acted to move the items to the new owner being "Del1". All of these are in their final states and so the machines are drawn in gray and marked as completed with the circled "C" at the top left. They can no longer change state or property values.

### **24:18 Event #17 - Invoice**

The seller finishes the transaction saying to the buyer "We're square, here is your invoice". The process overview moves to the "post-actualization" final state and the process detail moves to "completed" final state. The economic event "Pay1" changes to "event-complete" in fulfillment of the economic commitment "PlanPay1" which changes to "commitment complete" final state. The contract's wallet balance reduces to zero and the amount is disbursed to the seller and tax authority wallets. Finally, the contract records the duality of the actual stock flow delivery "Del1" with the actual payment "Pay1" of the stock flow allocation. The contract, too, moves to its completed final state.

### **25:24 Event #18 - Request for Quotation**

The second example in this scenario skips over the exchange of catalogues as this happened in the first example of the scenario. The buyer asks "Do you have the red bikes that I want?" through another request for quotation. This instantiates a new "Process2" pair of business process overview and detail state machines, a new offered "Contract2" economic contract pointing to a new proposed economic commitment "PlanDel2" for the delivery of 10 bikes on July 15.

### **26:06 Event #19 - Quotation**

The seller responds "Well, I have some of the bikes you want, but not all" with a Quotation document. This progresses the business process overview to the "negotiation" state and the detail to the "quotation received" state. The contract moves to the accepted state with the cost and tax added, and in the planned delivery the quantity is changed from 10 to 5. The

seller has redrawn the proposed contract parameters for what they can deliver.

### **26:45 Event #20 - Order**

The buyer responds "Okay, here is an order for red bikes" with an Order document. This instantiates the "PlanPay2" economic commitment, recorded in the contract, and referencing the new economic resource "Dai2", an allocation for 5000 Dai, the amount for the planned payment of 4000 Dai for the seller and 1000 Dai for the tax authority.

### **27:16 Event #21 - Order Response Simple**

The seller responds "Deal! The order is acceptable" with an Order Response Simple. This changes the state of the red bikes to be "reserved", and the planned delivery and payment to be "specified". The process overview moves to the "actualization" state and the detail to "order accepted" state.

### **27:43 Event #22 - Remittance Advice**

The buyer responds "Okay, here is the payment" which now reserves the "Dai2" allocation stock flow out of the buyer's wallet, reducing the wallet from 85000 Dai to 80000 Dai, and into the contract's wallet with 5000 Dai and the contract state to be "accepted". The "Pay2" economic event is instantiated with the stock flow allocation to be that "Dai2" allocation. The process overview remains in "actualization" and the detail state changes to the "paid" state.

At this point, the demonstration is over and the state machines reflect the current state of the incomplete business transaction.

### **28:45 Inside front cover - closing**

Going back to the previous illustration we have watched 22 of these business events occur. Here at the current or last event are the state and properties of all the state machines for the business transaction ontology primitives that were instantiated, three of which are brought out in focus from the last page of the PDF.

For the independent viewer of the OeDBTR, they can jump into the history of the business transaction by going to the associated page of the PDF file immutable store. Using other kinds of immutable storage, the access to the active state of the in-play primitives of the business transaction ontology can be projected into a perspective that is meaningful for that independent viewer.

### **29:43 Cover - closing**

Thank you for your interest in this illustration of a Distributed Business Transaction Repository immutable record of business transactions in accounting collaboration space.