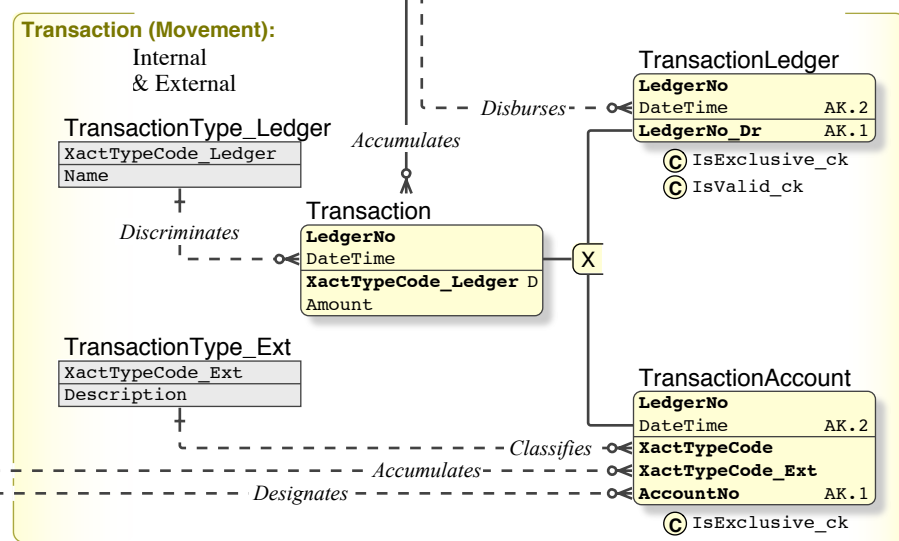
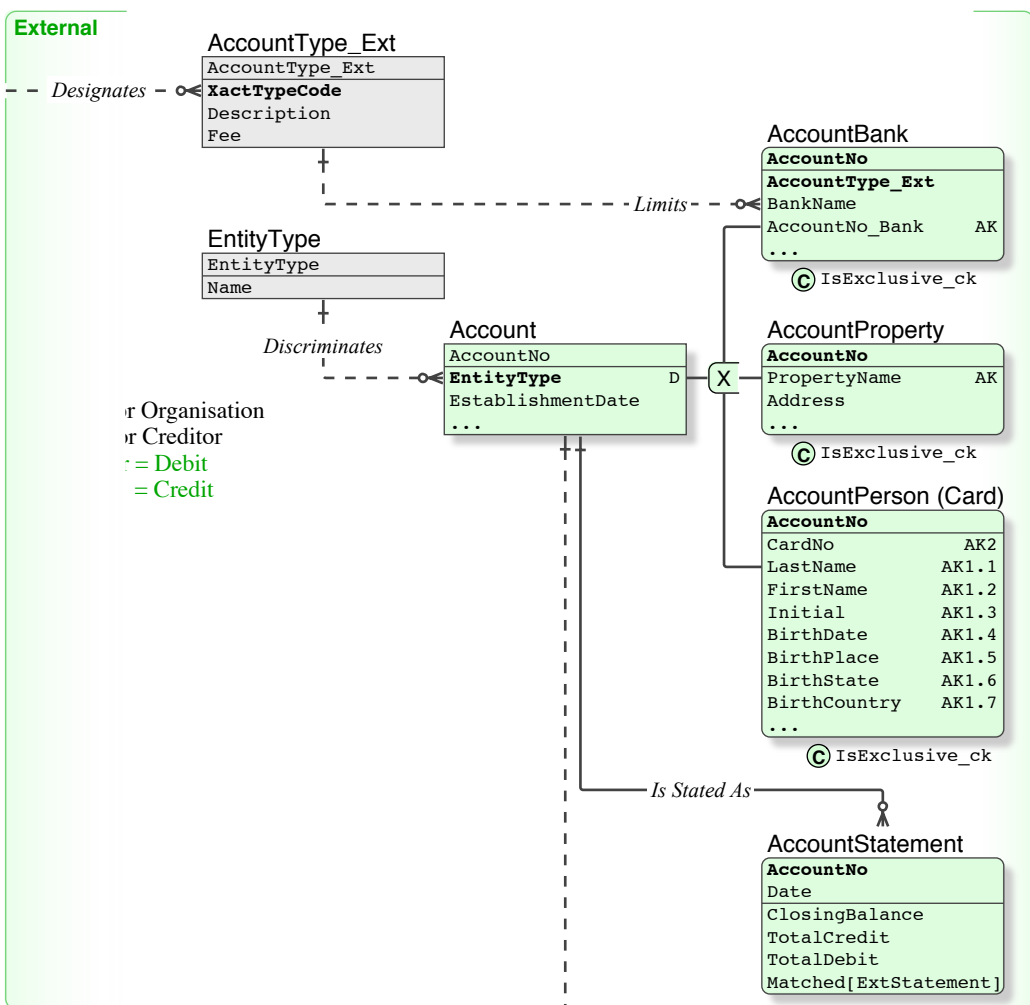
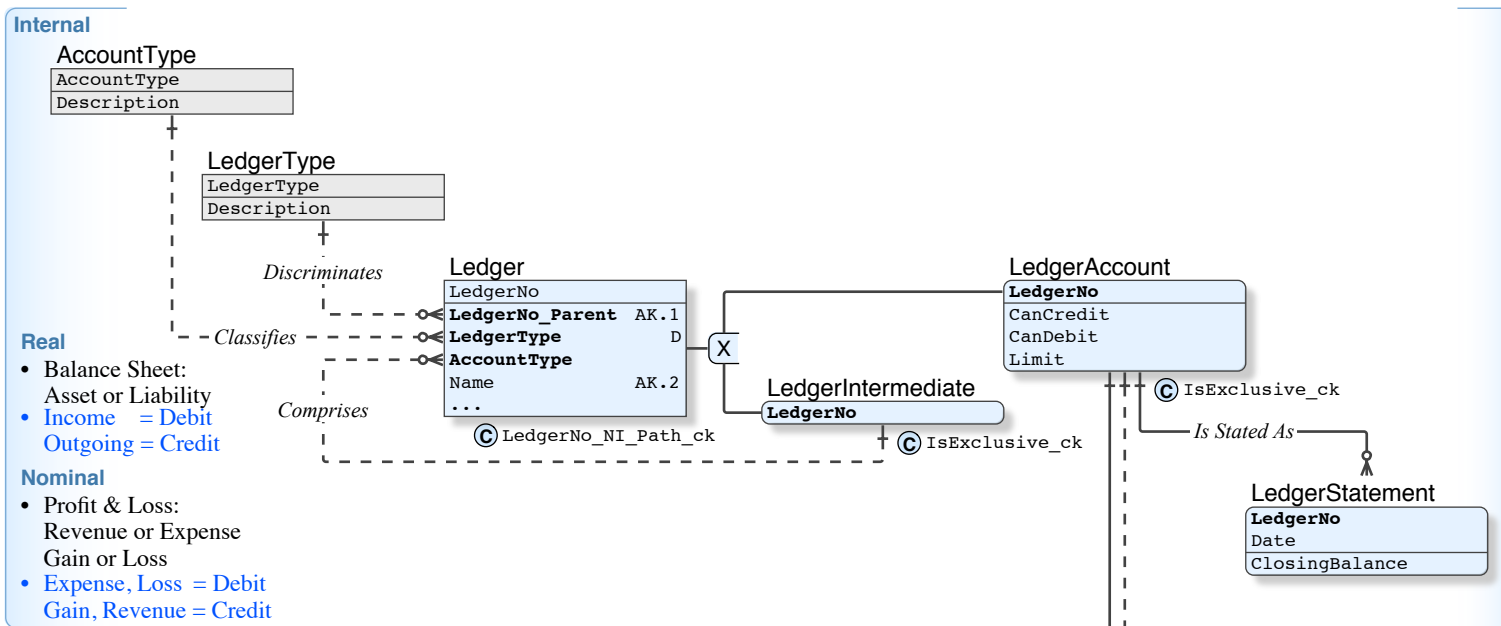
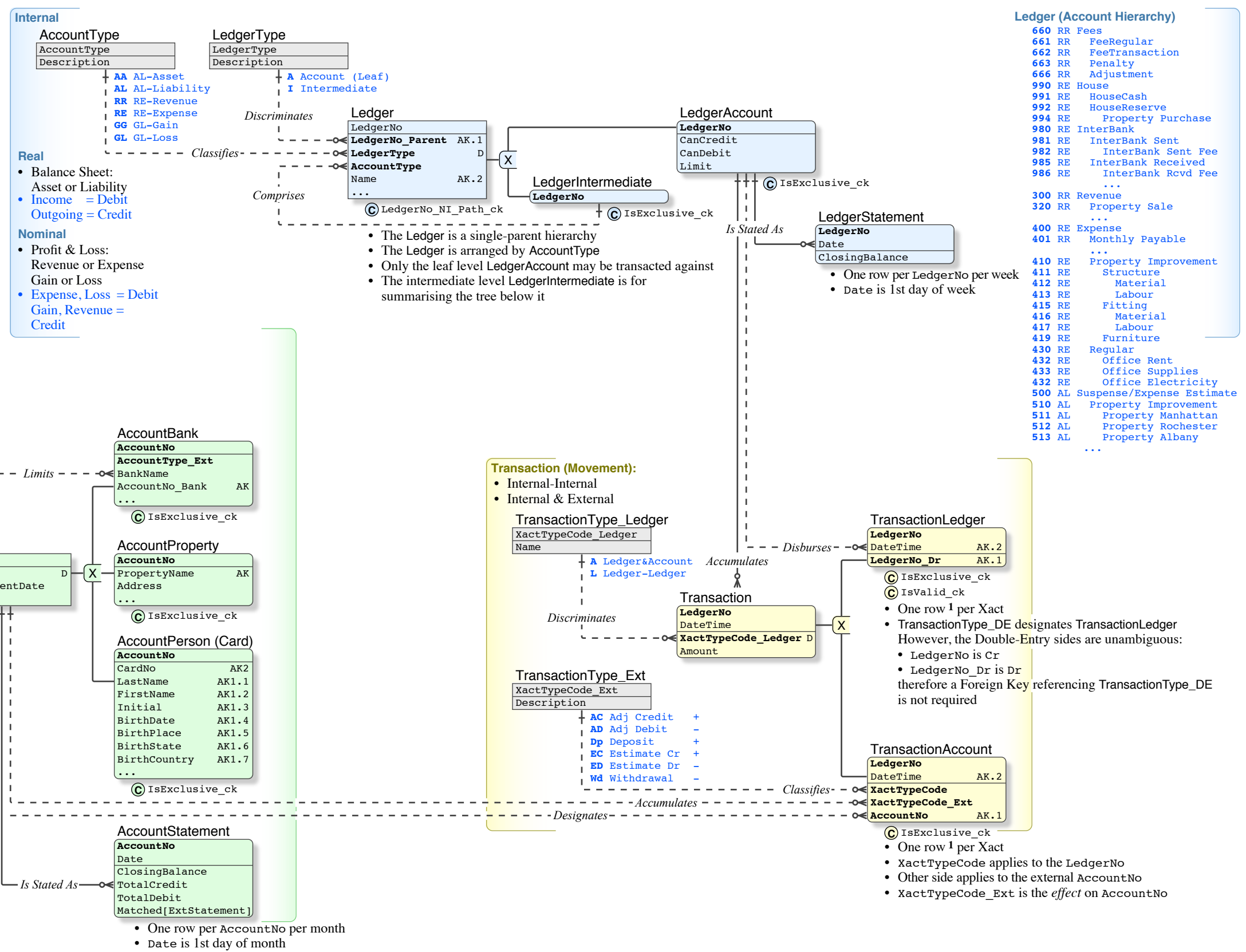


TransactionType_DE	
XactTypeCode	
Name	



Note

- 1 A single Basetype-Subtype pair, although physically split, is a single logical row.
- 2 Blanks are not stored in the column. Indentation is achieved in the SELECT code.
- 3 Transaction.DateTime is DATETIME, 3ms precision



Ledger (Account Hierarchy)

660	RR	Fees
661	RR	FeeRegular
662	RR	FeeTransaction
663	RR	Penalty
666	RR	Adjustment
990	RE	House
991	RE	HouseCash
992	RE	HouseReserve
994	RE	Property Purchase
980	RE	InterBank
981	RE	InterBank Sent
982	RE	InterBank Sent Fee
985	RE	InterBank Received
986	RE	InterBank Rcvd Fee
...		
300	RR	Revenue
320	RR	Property Sale
...		
400	RE	Expense
401	RR	Monthly Payable
...		
410	RE	Property Improvement
411	RE	Structure
412	RE	Material
413	RE	Labour
415	RE	Fitting
416	RE	Material
417	RE	Labour
419	RE	Furniture
430	RE	Regular
432	RE	Office Rent
433	RE	Office Supplies
432	RE	Office Electricity
500	AL	Suspense/Expense Estimate
510	AL	Property Improvement
511	AL	Property Manhattan
512	AL	Property Rochester
513	AL	Property Albany
...		

Context

- There are three levels of Q & A that are involved. Nevertheless, I have attempted to maintain the whole in three increments
- Following StackOverflow, each Answer is limited to the context of the Question. Each Answer includes a data model

- 1 [Derived account balance vs stored account balance for a simple bank account](#)
- 2 [Relational model for double entry accounting](#)
- 3 [Relational model for double entry accounting with Job Costing](#)

This third level data model provides the full Accounting context:

- the Ledger is a fully expanded single-parent hierarchy Refer to [Hierarchy](#) for a full definition of Relational Hierarchies
- Transaction is further Normalised
- Account is further Normalised
- All CONSTRAINTS are shown

```

(C) AccountBank.IsExclusive_ck
CHECK ( Account_IsExclusive_fn( AccountNo, "B" ) = 1 )
(C) AccountProperty.IsExclusive_ck
CHECK ( Account_IsExclusive_fn( AccountNo, "P" ) = 1 )
(C) LedgerAccount.IsExclusive_ck
CHECK ( Ledger_IsExclusive_fn( LedgerNo, "A" ) = 1 )
(C) LedgerIntermediate.IsExclusive_ck
CHECK ( Ledger_IsExclusive_fn( LedgerNo, "I" ) = 1 )

(C) LedgerNo_NI_Path_ck
CHECK LedgerNo NOT IN Ledger_Path_fn( LedgerNo_Parent )
(C) TransactionAccount.IsExclusive_ck
CHECK ( Transaction_IsExclusive_fn( LedgerNo, "A" ) = 1 )
(C) TransactionLedger.IsValid_ck
CHECK ( TransactionLedger_IsValid_fn( LedgerNo, LedgerNo_Dr ) = 1 )
-- check LedgerNo NE LedgerNo_Dr
-- check LedgerNo & LedgerNo_Dr are valid re CanCredit/CanDebit
    
```

Table Type

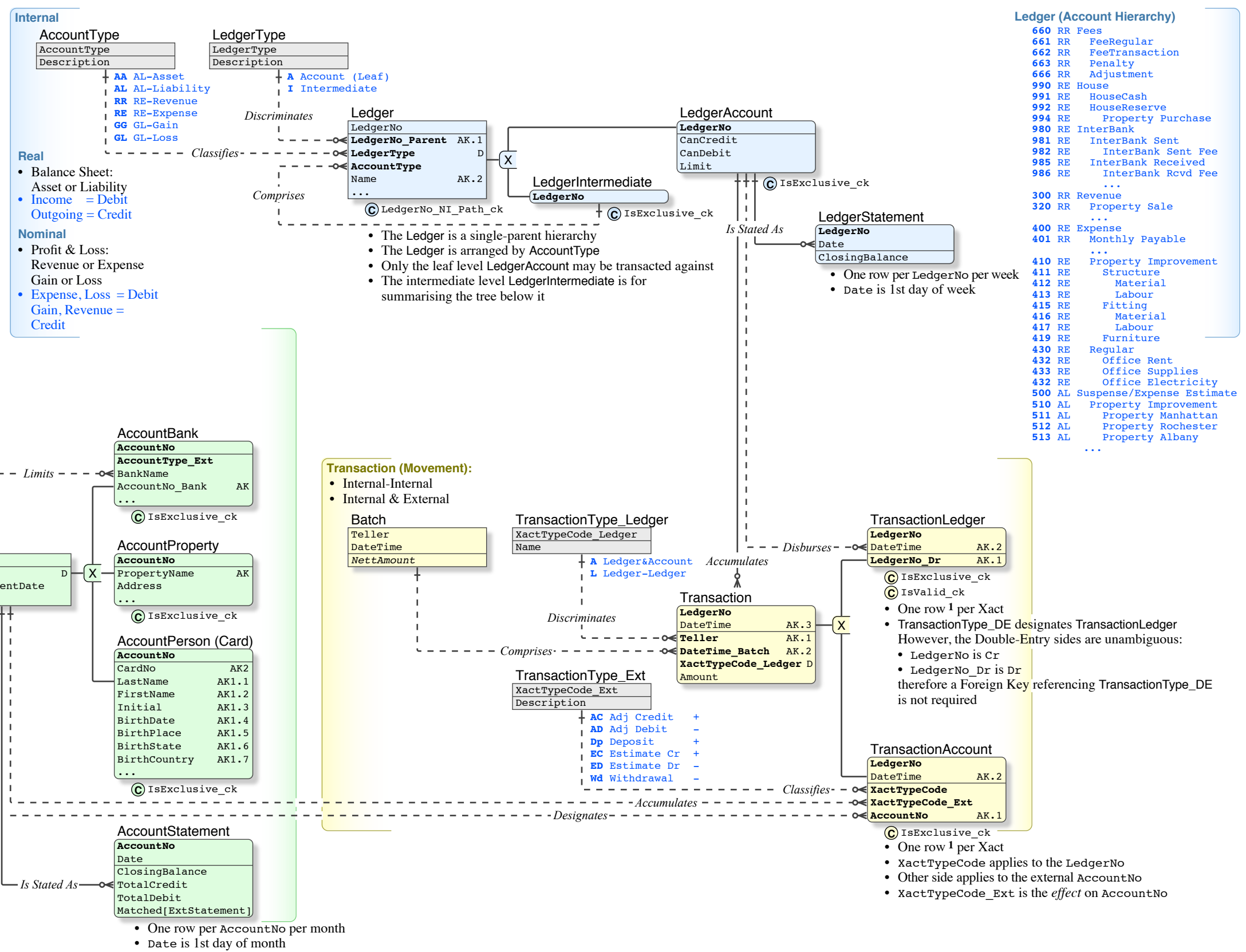
- Ledger (Internal)
- Account (External)
- Transaction

Internal PrimaryKey Attribute
External PrimaryKey Attribute

IDEFIX Notation

Note

- 1 A single BaseType-Subtype pair, although physically split, is a single logical row.
- 2 Blanks are not stored in the column. Indentation is achieved in the SELECT code.
- 3 Transaction.DateTime is DATETIME, 3ms precision
- 4 Batch.DateTime is SMALLDATETIME, 1min precision



Context

- There are three levels of Q & A that are involved. Nevertheless, I have attempted to maintain the whole in three increments
- Following StackOverflow, each Answer is limited to the context of the Question. Each Answer includes a data model

- 1 [Derived account balance vs stored account balance for a simple bank account](#)
- 2 [Relational model for double entry accounting](#)
- 3 [Relational model for double entry accounting with Job Costing](#)

This third level data model provides the full Accounting context:

- the Ledger is a fully expanded single-parent hierarchy Refer to [Hierarchy](#) for a full definition of Relational Hierarchies
- Transaction is further Normalised
- Account is further Normalised
- All CONSTRAINTS are shown

```

(C) AccountBank.IsExclusive_ck
CHECK ( Account_IsExclusive_fn( AccountNo, "B" ) = 1 )

(C) AccountProperty.IsExclusive_ck
CHECK ( Account_IsExclusive_fn( AccountNo, "P" ) = 1 )

(C) AccountPerson.IsExclusive_ck
CHECK ( Account_IsExclusive_fn( AccountNo, "p" ) = 1 )

(C) LedgerAccount.IsExclusive_ck
CHECK ( Ledger_IsExclusive_fn( LedgerNo, "A" ) = 1 )

(C) LedgerIntermediate.IsExclusive_ck
CHECK ( Ledger_IsExclusive_fn( LedgerNo, "I" ) = 1 )

(C) LedgerNo_NI_Path_ck
CHECK LedgerNo NOT IN Ledger_Path_fn( LedgerNo_Parent )

(C) TransactionAccount.IsExclusive_ck
CHECK ( Transaction_IsExclusive_fn( LedgerNo, "A" ) = 1 )

(C) TransactionLedger.IsExclusive_ck
CHECK ( Transaction_IsExclusive_fn( LedgerNo, "L" ) = 1 )

(C) TransactionLedger.IsValid_ck
CHECK ( TransactionLedger_IsValid_fn( LedgerNo, LedgerNo_Dr ) = 1 )
-- check LedgerNo NE LedgerNo_Dr
-- check LedgerNo & LedgerNo_Dr are valid re CanCredit/CanDebit
    
```

Table Type

- Ledger (Internal)
- Account (External)
- Transaction

Internal
PrimaryKey
AlternateKey

External
PrimaryKey
AlternateKey

Attribute

IDEFIX Notation