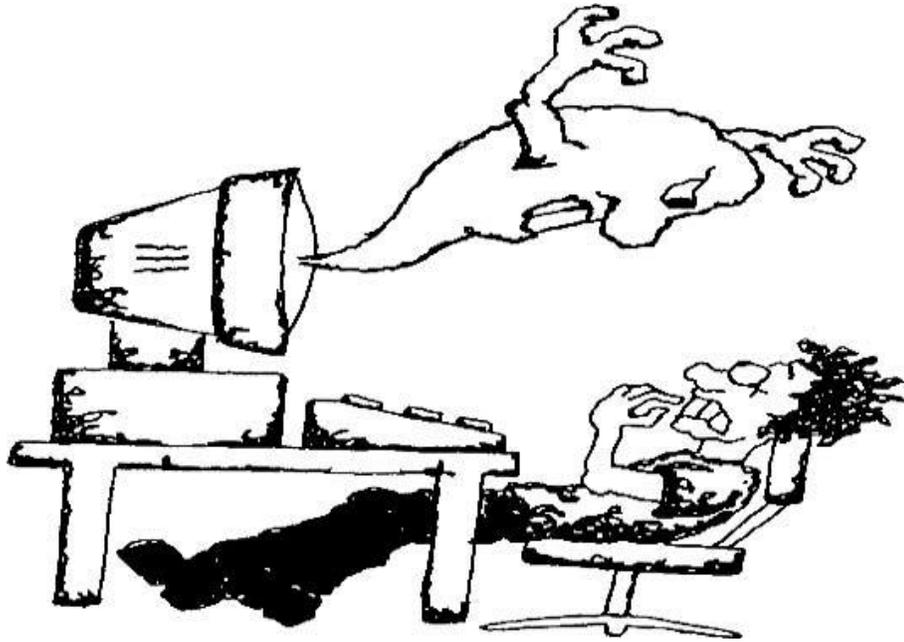




CRC – Not as Scary as you think



Dan Mitchell and Susan Houniet (and Marv Stone)

Agenda

1. What is CRC?
2. Traditional method to make schema changes
3. Online Schema Changes
4. Futures



What Does CRC Stand For?

- A. Chemical Rubber Company**
- B. Civil Rights Center**
- C. Colo-Rectal Carcinoma**
- D. Cooperative Research Center**
- E. Cyclic Redundancy Check**
- F. College Rugby Championship**
- G. Camp Robinson Crusoe**
- H. Crisis Response Cell**

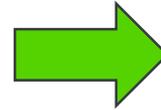
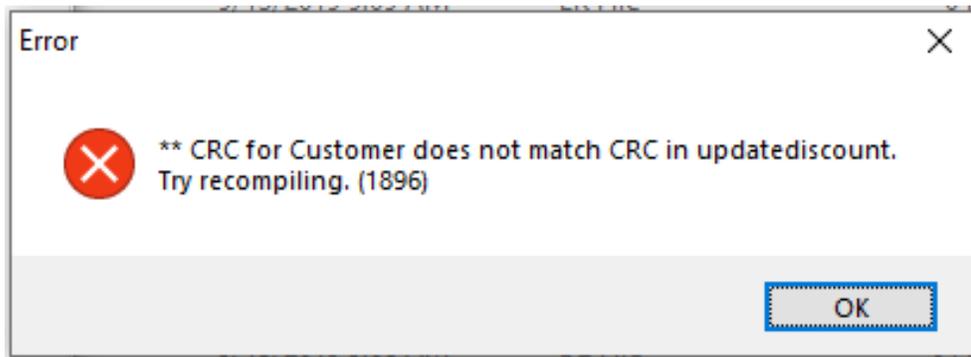


What Does CRC Stand For?

- A. Chemical Rubber Company
- B. Civil Rights Center
- C. Colo-Rectal Carcinoma
- D. Cooperative Research Center
- E. Cyclic Redundancy Check**
- F. College Rugby Championship
- G. Camp Robinson Crusoe
- H. Crisis Response Cell



Why Does CRC Matter?



Why CRC?

- Ensures that program running only has references to database objects (**Tables, Indexes and LOBs**) where definitions have remained unchanged since the last compile.
- Prevents ungraceful runtime errors and writes to the database with internal inconsistencies
 - Could occur when the application software tries to address a non-existent or unrecognized data structure.
- This is done by storing CRC values for tables and indexes into the compiled code. (LOB fields are calculated into the table CRC).



What Makes Up the CRC

- **For tables**, the CRC is stored if there is a static reference to the table or one of its fields in the code.
- **For indexes**, the CRC is stored if the index is used by a static QUERY, FOR EACH or FIND construct.
- A **dynamic reference** to a static buffer will require recompiling when the schema changes.



Changes that **ALWAYS** Require Recompile to Avoid CRC Error - TODAY

■ **Tables**

- Changing the name of the table or index
- Deleting a field in the table

■ **Fields**

- Changing the name, data type, decimal precision, extent(array), or case sensitivity
- Changing foreign database metadata in a schema-holder database for Dataservers (foreign database field type, length, offset, or case sensitivity)
- `_record._fieldpos`

■ **Indexes**

- Changing the index name, uniqueness requirement or abbreviated option.
- Adding, removing or changing the order of the indexed fields
- Changing the ascending/descending order of an indexed field



Traditional method to make schema changes

Traditional method to make schema changes.

1. Connect to dev.db
 - Make schema changes in the Data Dictionary
2. Make code changes, compile and deploy
3. Connect to dev.db and Prod.db
 - Create delta.df
4. Apply delta.df to the Prod.db
5. Start Prod.db database or clients reconnect



What is an Incremental delta.df file?

- When you make changes to your database schema you can create a **delta.df** file that shows the differences between the old and new database schema.
- Examples
 - Add a new field/column to an existing table
 - Add a new index to an existing table
 - Delete a table

Admin > Dump Data and Definitions > Create Incremental .df File.

How much downtime is needed?



Green = online, Red = Schema Lock

For any version prior to OpenEdge 10.2A, a Schema lock is needed to make changes to the database (single user mode).



Time for Change

Vision

There is no reason your OpenEdge application should ever stop running

Availability Goal

99.999

How Can You Minimize Downtime when Deploying Schema Changes?



Online Schema Changes

Things you **CAN** change online!!!

Schema Changes Online, OE 10.2A+

■ Add New **Tables**

- All objects associated with the table must be made in the same transaction, including add any fields, indexes or database triggers belonging to the *new table*.
- Active indexes can only be created online if they are created on a newly created table within the same transaction as the CREATE TABLE.

■ Add *new **fields*** to an existing table, but no Assign triggers. The field record position value should not conflict with the record position of existing fields for online schema changes.

■ Add *new **inactive indexes*** to an existing table.

■ Add New **Sequences**.

Things you **CAN** change online!!!

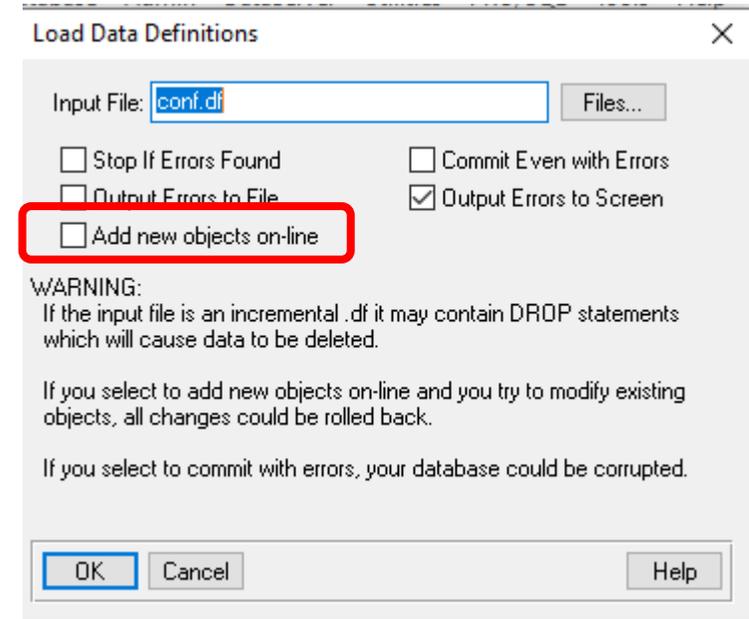
Non-structural Schema Field Changes Online, OE 12.1+

- `_Field._Col-label-SA`
 - `_Field._Col-label`
 - `_Field._Fld-case`
 - `_Field._Format-SA`
 - `_Field._Format`
 - `_Field._Help-SA`
 - `_Field._Help`
 - `_Field._Label-SA`
 - `_Field._Label`
 - `_Field._Valexp`
 - `_Field._Valmsg-SA`
 - `_Field._Valmsg`
 - `_Field._View-As`
 - `_Field._Width`
 - `_File._File-Label-SA`
 - `_File._File-Label`
 - `_File._Valexp`
 - `_File._Valmsg-SA`
 - `_File._Valmsg`
-
- *Datatype from integer to int64*

Applying a delta.df On-line

1. Open Data Administration Tool
2. Connect to the database
3. From the **Admin -> Load Data and Definitions** menu, select **Data Definitions (.df file)**...
4. On the Load Data Definitions Dialogue, turn on the "**Add new objects on-line**" toggle-box and choose the OK button to load online schema changes into the Database

Note: If the .df to be applied contains any other action that requires a schema lock, Progress will issue an error and all changes will be rolled back.



Applying a delta.df On-line Using Code

```
Session:SCHEMA-CHANGE = "NEW OBJECTS".
```

```
Run product/load_df.p("delta.df").
```

```
Session:SCHEMA-CHANGE = "".
```

Online Index Activation

- Cannot add primary index using this
- Must have an active primary key or use “useindex” to specific a valid, active index - works better is primary key or “useindex” is unique because we need to visit every record in the table
- Should run IDXFIX prior to IDXACTIVATE to pre-build the index keys
- The must be no users with a schema timestamp that is earlier than the schema timestamp of the index
- In OpenEdge 11.7 when Database Client Notification (-usernotifytime) is enabled on database startup, IDXACTIVATE waits until all connected clients respond to the notification, and then proceeds without the user having to take action.
- Takes a long time and creates a lot of log entries

<https://knowledgebase.progress.com/articles/Article/Can-inactive-indexes-be-built-online-000060410>



How much downtime is needed?

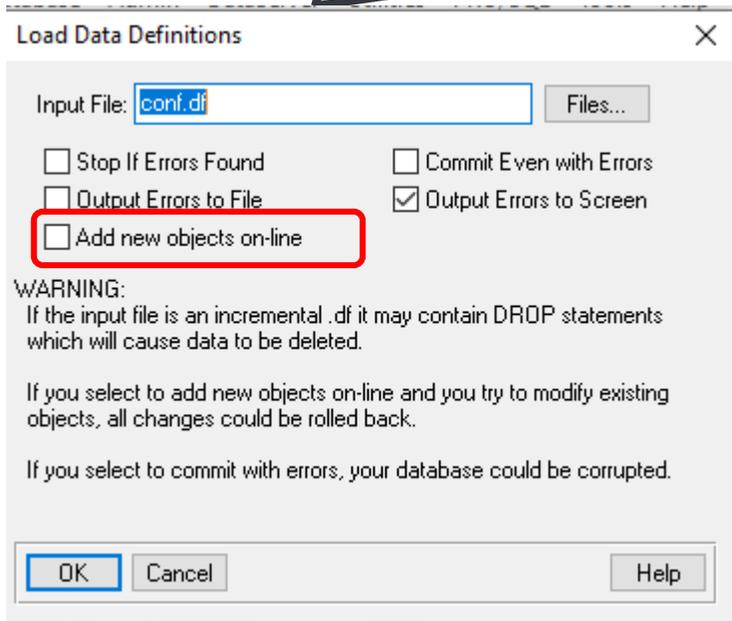


Green = online, Red = Schema Lock

For any version prior to OpenEdge 10.2A, a Schema lock is needed to make changes to the database (single user mode).



How much downtime is needed?



*Note: If the .df to be applied contains any other actions that requires a **schema lock**, Progress will issue an error and **all** changes will be rolled back.*



Now schema changes can be done online

- What happens to your rcode...
 - When you add a table?
 - When you add a field?
 - When you add a index?
 - When you change field properties?

It just works

The old column label continues to appear until the r-code is recompiled.

This does not affect how the business logic executes.



Futures

Beyond OE12.1

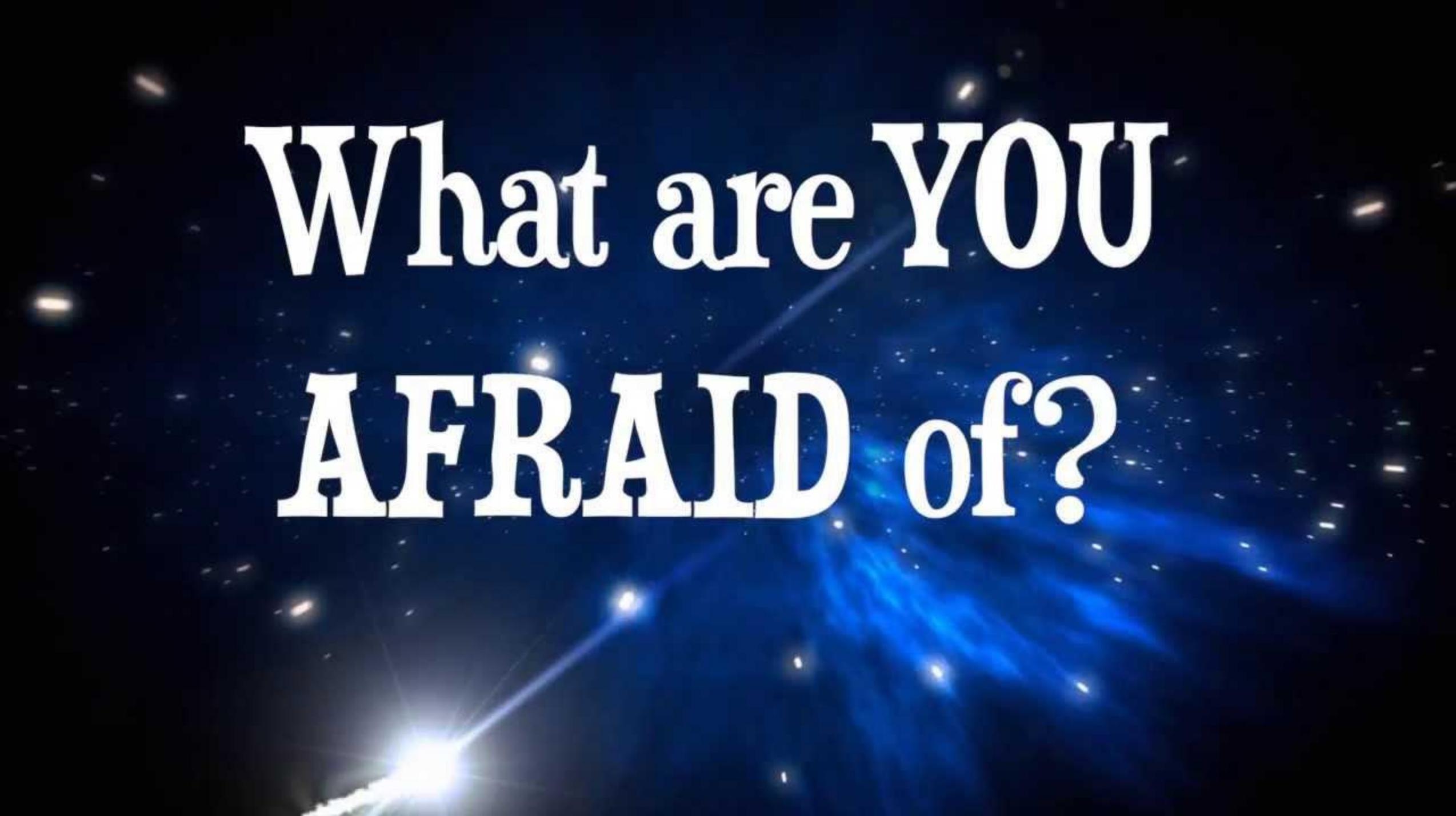
- ✓ *Database triggers (CUD)*
- ✓ *Field Names (rename, drop)*
- ✓ *Table Name (rename, drop)*
- ✓ *Index (drop inactive, deactivate)*
- ✓ *Sequence (rename, drop)*

Automated Multi-Stage .df Application

- Currently, create delta.df and apply it
- Some customers scan for operations that can't be done online and strip them out
- Proposal is to change the dictionary so that all .df files are created in sections
- For example...
 - **Pre-deploy section** – before you deploy application while old version of application is running
 - **Database trigger changes** – changes to trigger layout
 - **Post deploy operations** – things that will disrupt current version of application but still online
 - **Off-line changes** – changes that require a maintenance window to be done while the database is effectively running in single-user mode.

■





**What are YOU
AFRAID of?**

Useful sessions

- Session 542: Let's Talk Five 9's – 4:45 on Tuesday in Pemigewasset Room with Tim Sargent and Raghu Rangan
- Session 318: Index Maintenance Utilities... Already happened so review presentation slides when available (with Paul Koufalis)

QUESTIONS?

