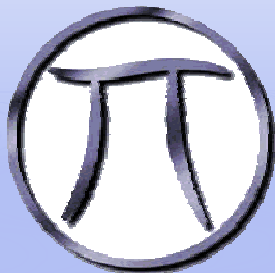


# VS01– Global User Exits und Resource Management Interface CICS

Martin Trübner, H-System

Heinz Peter Maassen, Lattwein GmbH





# Agenda

- p User Exits im CICS TS
  - n Global User Exit
  - n Task Related User Exit
  - n User replaceable Modules
- p Modify EIBDATE Routine im XEIOUT
- p Security Check im XPCFTCH EXIT
- p TRUE Exit Programming mit Martin



# Was bedeutet Global User Exit?

- Der globale User Exit Punkt – entspricht genau einer vordefinierten Stelle in einem CICS Modul oder CICS Domäne, an der CICS TS die Kontrolle an ein Global User Exit Programm übergibt. CICS TS erhält nach Durchführen des Programms wieder die Kontrolle. Man braucht keine globalen User Exits zu benutzen, wenn man sie nicht benötigt. Jedoch man kann diese benutzen, um CICS TS an seine Bedürfnisse anzupassen.
- Hier einige Beispiele zu den globalen User Exits:

Exit Name	Module oder Domäne	Wo oder Wann aufgerufen
XEIIIN	EXEC Interface Program	Vor Ausführung eines EXEC CICS Befehls.
XEIOUT		Nach Ausführung eines EXEC CICS Befehls.
XKCREQ	Task Control Program	Vor einem ENQUEUE oder DEQUEUE Request.
XPCABND	Program Control Program	Bevor ein Dump Call ausgeführt wird
XPCFTCH		Bevor einem Anwendungsprogramm die Kontrolle übergeben wird
XPCHAIR		Bevor einer HANDLE ABEND Routine die Kontrolle übergeben wird



## Was bedeutet: **Global User Exit?**

- z Jeder Globale User Exit Punkt hat einen eindeutigen Namen auch Identifier genannt, und befindet sich an einer geeigneten Stelle in den CICS Modulen oder Domänen um zusätzliche Aufgaben zu erledigen. Der Exit Punkt XPCFTCH in der Programm Domain wird immer dann aufgerufen, bevor die Kontrolle an dieses Programm übergeben wird. Das ist eine ideale Stelle um beispielsweise einer eigenen Security Check Routine prüfen zu lassen, ob das aufgerufene Programm vom derzeitigen User benutzt werden darf oder nicht.
- z Globale User Exit Unterstützung wird von CICS automatisch bereit gestellt. Trotzdem gibt es einige Regeln die beschreiben nach welchen Konventionen das jeweilige EXIT zu schreiben ist. Es gibt Standard Parameter die einem EXIT Modul übergeben werden und ebenso gibt es Standard Return Informationen die der aufrufenden CICS Routine Werte zurück übermitteln.





## Was bedeutet: **Global User Exit?**

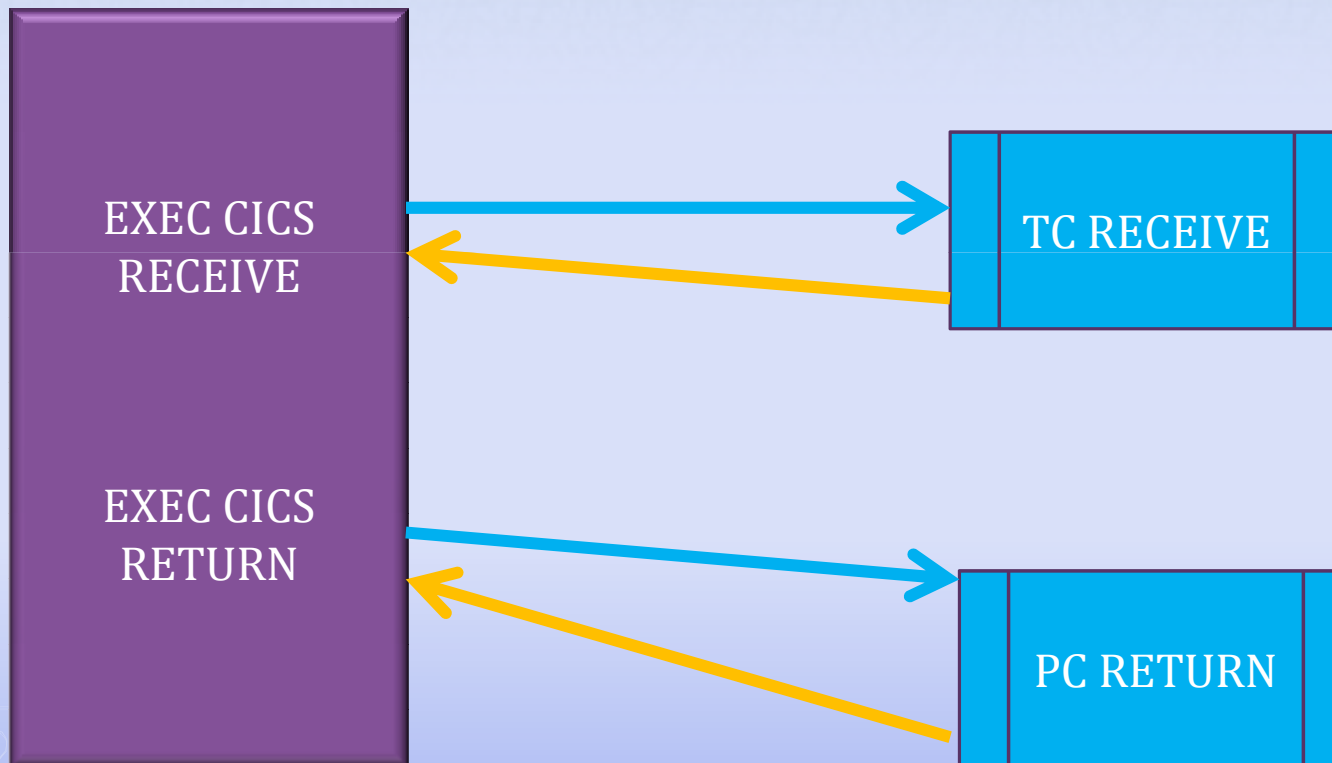
- z Da die meisten User Exit Programme so arbeiten als wären sie Bestandteil eines CICS Moduls ergeben sich Einschränkungen zur Benutzung der CICS Schnittstellen (Commands). Die meisten Global User Exit Programme können keine EXEC CICS Commands benutzen. Dafür stehen für bestimmte CICS Dienste die Befehle des Exit Programming Interfaces (XPI) zur Verfügung.
- z **Achtung:** Es wird weder eine Source noch Objekt Kompatibilität zu kürzlich erschienenen CICS Versionen garantiert. Jedoch werden alle Änderungen in den entsprechenden Handbüchern dokumentiert sein.



# CICS Global User Exit inaktiv

p Anwendung

CICS Module

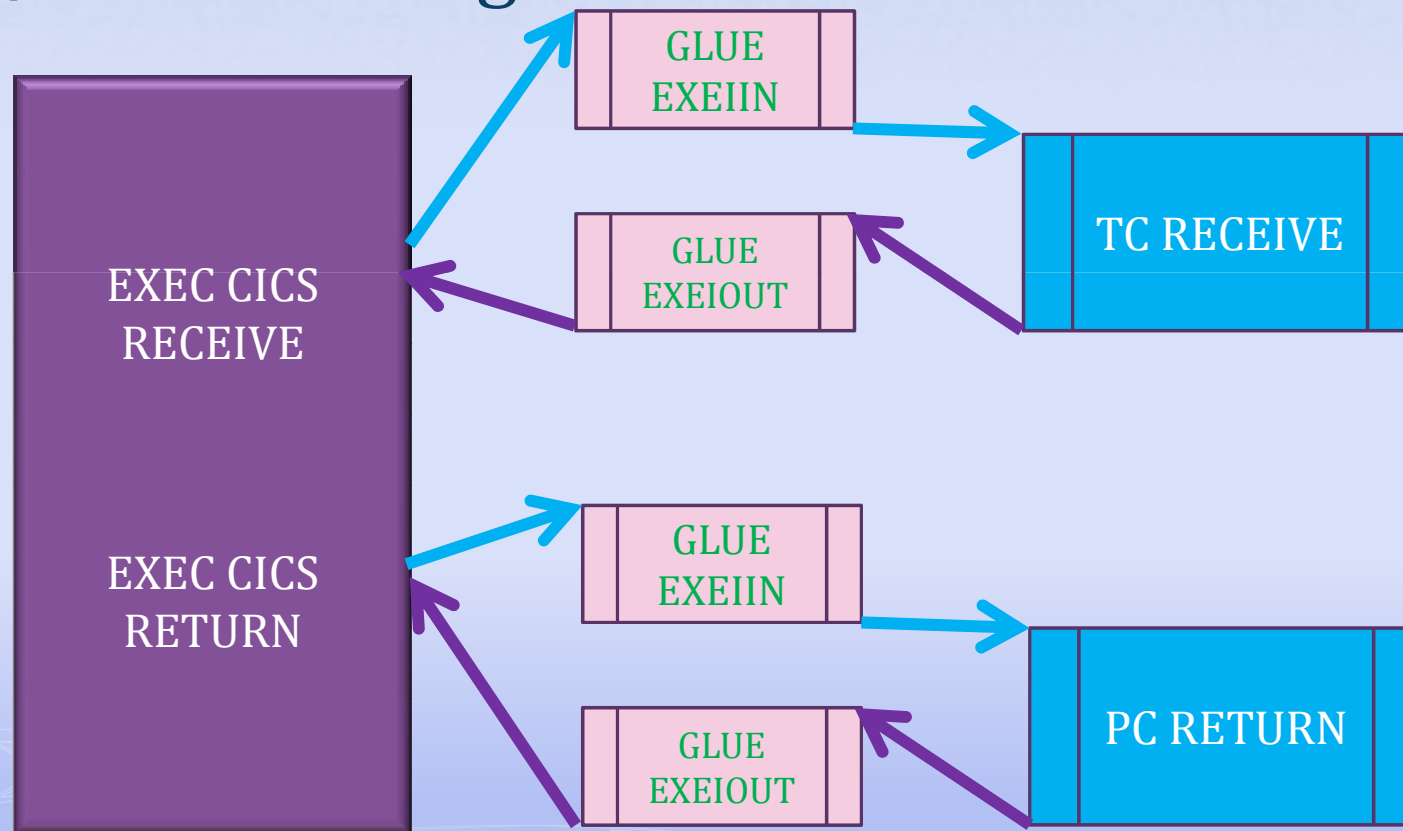




# CICS Global User Exit aktiv

p Anwendung

CICS Module





# GLUE: XEIIIN

Das Exit XEIIIN wird aufgerufen, bevor ein EXEC CICS Command ausgeführt wird

## Exit-Übergabe Parameter

<b>UEPARG</b>	Adresse der EXEC Command Parameter Liste
<b>UEPEXECB</b>	Adresse der SYSTEM EIB.
<b>UEPUSID</b>	Adresse der 8 stelligen UserID.
<b>UEPPGM</b>	Adresse des 8stelligen Programmnamens
<b>UEPLOAD</b>	Adresse des Anwenderprogramms.
<b>UEPRSA</b>	Adresse der Register Save Area des Anwendungsprogramms. Hier sind die Registerinhalte zum Zeitpunkt des EXEC CICS Befehles hinterlegt.

## Return Codes

<b>UERCNORM</b>	Verarbeitung fortsetzen.
<b>UERCBYP</b>	Den Befehl übergehen (bypass)
<b>UERCPURG</b>	Task wird depurged während des XEIIIN Calls

**XPI Calls** können alle verwendet werden

**API und SPI Commands** dürfen nicht verwendet werden.





## XPI Commands zum GLUE

- p Es gibt 3 SPI Commands für Global User Exits
  - n ENABLE EXIT Aktivieren des Exits evtl.  
mit einer Globalen Work Area
  - n EXTRACT EXIT            Adressieren GWA
  - n DISABLE EXIT            Deaktivieren des Exits



# Beispiel: EXIT XEIOUT aktivieren

```
EXEC CICS ENABLE PROGRAM('MYEXOUT') EXIT('XEIOUT') X  
      GALENGTH(2000) ENTRYNAME('MXO') RESP(MRESP)
```

```
EXEC CICS EXTRACT EXIT PROGRAM('MYEXOUT') ENTRYNAME('MXO') X  
      GASET(R5) GALENGTH(GL) RESP(MRESP)
```

Der erste Command: ENABLE PROGRAM:

- z Lädt das Exit Programm in die CICS Partition
- z Definiert den Entry des EXIT Programms für CICS
- z Holt die Workarea für dieses Exit Programm
- z Stellt bei globalen Exits die Verbindung zwischen dem Exit und dem CICS Modul her.
- z Bei TRUE wird das Exit ab nun bei jeder Task gestartet
- z Aktiviert somit das EXIT



# EXEIOU

- ž Dieses User Exit wird nach Ausführung eines Q CS Commands ausgeführt
  
- ž Was kann man hier anstellen?
  - Man hat Zugriff auf den DFHEIB Storage
  - Man kann z.B. eine Terminal ID abfragen
  - Oder die Transaktion
  - Oder Datum und Uhrzeit
  - Und kann der Anwendung genau diese Informationen verändern und andere Werte hier setzen
  - Z.B. statt Datum 20.04.2010 das Datum 01.07.2010.



# EXEIOU

- ž Man kann natürlich die Manipulationen auch in Abhängigkeit von einer Terminal ID machen
- ž Beispiel: Nur wenn die Terminal ID = TT\* ist soll das Datum auf ein anderes Datum geändert werden
- ž Wichtig ist allerdings: CICS darf davon nichts merken.





## EXIT

- ⌘ Dieser User Exit wird **vor** Ausführung eines CLS Commands ausgeführt
- ⌘ Dieser Exit kann man verwenden um die Werte, die man verändert hat auf die Ursprungswerte zurückzustellen
- ⌘ Also EIBDATE = 20.04.2010 statt 1.07.2010
- ⌘ Auch hier kann man die Terminal ID abfragen – Nur wenn TERMINAL mit TT.. beginnt.



## Vorteile

- ž Globale User Exits können global den Ablauf von Programmen beeinflussen, auch wenn man keinen Source Code zur Verfügung hat
- ž GLUE können auf eine gemeinsame GWA zugreifen. Hier können globale Definitionen hinterlegt werden
- ž Können mit ENABLE aktiviert und mit DISABLE deaktiviert werden – Kein CICS Neustart erforderlich.



# Nachteil

- ž Geht nur mit HLASM.
- ž Keine Kompatibilität - Garantiefür zukünftige CICS T Versionen.
- ž Kleine Fehler können das gesamte System beeinflussen – bis zum Stillstand.



# User Replaceable Modules.

- ž User Replaceable Modules hat jeder im Einsatz – auch wenn er diese nie modifiziert hat.
  
- ž Bekannt sind die folgenden:
  - Program Error Programm
  - Transaction Restart Programm
  - Terminal Error Programm
  - Node Error Programm
  - Programm für Austausch von Terminals
  - Programm für Austausch von Programmen
  - Und andere





# Task Related User Exits

- ž TRUE werden auch von IBM benutzt um im CICS fehlende Anbindungen an nicht unterstützte Subsysteme zu ermöglichen
  
- ž 2 Beispiele:
  - 1.) DB2 VSE. Alle DB2 Befehle werden über ein TRUE abgearbeitet – mit CIRB wird das TRUE gestartet.
  - 2.) Das EZASMI TCP/IP Programming API wird seit z/VSE 4.1 über EZATSTART(oder PLT) im CICS/TS aktiviert.