

## Transcript - Group 3 "Modelers", Interview 2

I ... Interviewer (BLINDED)

B ... Expert

(Unv.)... Incomprehensible passage

(...) ... Pause longer than 3 sec.

( ) ... Comment

// ...// ... Speaker overlap

### Transcript

1 1. I: Okay. Aufnahme läuft. #00:00:04-2#

2 2. B: Okay. #00:00:05-7#

3 3. I: Hallo und danke, dass Sie sich Zeit nehmen, um mit mir dieses Interview durchzuführen.

4 [REDACTED]  
5 Möchten Sie sich vielleicht kurz vorstellen und Ihre Verbindung zu BPMN beziehungsweise zur  
6 Verfahrenstechnik, Fertigungstechnik oder Prozessmodellierung erklären? Ich möchte Sie bitten,  
7 dabei nicht Ihren Namen zu nennen, sondern nur die folgenden Informationen. Und zwar.  
8 Berufsbezeichnung und Umschreibung des Arbeitgebers, Basis der Expertise zum  
9 Forschungsthema, Ausbildung beziehungsweise fachlicher Hintergrund und Berufserfahrung.  
10 Bitte. #00:00:49-7#

11 4. B: Ja, sehr gerne. [REDACTED]

12 [REDACTED]  
13 [REDACTED]  
14 [REDACTED]  
15 [REDACTED]  
16 [REDACTED]  
17 [REDACTED]  
18 [REDACTED]  
19 [REDACTED]  
20 [REDACTED]  
21 [REDACTED] Daher kommt auch noch ein bisschen Expertise.  
22 Genau. #00:01:58-7#

23 5. I: Okay. Dankeschön. Dann zum Forschungsthema. Unsere Forschung konzentriert sich auf die  
24 Entwicklung einer Methodik, um kontinuierliche Prozesse in BPMN darzustellen und sie in einer  
25 Workflow Engine ausführbar zu machen. Für diese Aufgabe haben wir an BPMN-Erweiterungen  
26 für kontinuierliche Prozesse gearbeitet. Warum kontinuierliche Prozesse? Weil diskrete Prozesse  
27 bereits in anderen Forschungsarbeiten behandelt wurden und nicht die gleichen Schwierigkeiten  
28 bei der korrekten Darstellung mittels BPMN aufweisen. BPMN ist bereits ein weit verbreiteter  
29 Standard im Business Process Management und hat seinen Weg in die Fertigung gefunden.  
30 Diskrete Fertigungsprozesse können bereits mit BPMN 2.0 modelliert werden. Im Grunde wollen  
31 wir eine Methodik einführen, um solche Prozesse so darzustellen, dass sie von jeder Person in  
32 einem Unternehmen, vom Ingenieur bis zum Manager, verstanden werden können. Dies könnte  
33 durch die Verwendung dieser Notation erreicht werden. Ein weiterer Vorteil ist auch, dass es  
34 bereits eine Reihe von Workflow Engines gibt. Anwendungen, die die Ausführung dieser  
35 Prozessmodelle auf der Grundlage der für jedes Symbol implementierten Logik ermöglichen. Wir  
36 arbeiten mit einer webbasierten Anwendung, die erweiterbar ist und mehrere  
37 Kommunikationsschnittstellen implementiert hat. Ein weiterer Vorteil ist daher die  
38 Interoperabilität in diesem Zusammenhang im Vergleich zu anderen proprietären starren  
39 Software-Anwendungen. Wir wollen herausfinden, ob diese Technik auch für die  
40 Implementierung von digitalen Abbildern eingesetzt werden kann. Da digitale Abbilder dazu  
41 dienen, ein physikalisches System oder einen Prozess in digitaler Form darzustellen, meist  
42 anhand von Daten oder mathematischen Modellen, mussten wir einen Weg finden den Ablauf

von kontinuierlichen Prozessen, wie sie aus der Prozessindustrie bekannt sind, darzustellen. Aus diesem Grund haben wir uns auf die Modellierung von Regelkreisen konzentriert. Die Prozessmodelle sollen durch BPMN für Personen mit unterschiedlichem Hintergrund leicht verständlich sein. Die Interviews werden geführt, um herauszufinden, wie Prozess- und Regelungstechnik und Techniken aus der Business Process Modellierung kombiniert werden können, und wie erste Ergebnisse von Experten wie Ihnen wahrgenommen werden. Außerdem wollen wir herausfinden, ob es Schwachstellen gibt, die von Experten identifiziert werden und wie wird diese beseitigen können. Und bevor wir jetzt zum Fragenteil kommen, würde ich gerne noch drei Begriffe vorab klären. Und zwar. Einmal der Begriff digitaler Zwilling. Es gibt verschiedene Methoden Dinge aus der echten Welt, zum Beispiel echte Maschinen, zu simulieren. Teils merkt man aber, dass es mehr Parameter brauchen würde, als bei normalen Simulationsmethoden, um eine Maschine vollkommen so abzubilden, wie sie sich in der Realität verhält. Bei einem digitalen Zwilling wird versucht, möglichst nahe an das reale Verhalten einer Maschine oder anderer Objekte heranzukommen. Das soll dazu führen, dass wenn etwas getriggert wird bei einer echten Maschine, der digitale Zwilling das gleiche oder ein möglichst ähnliches Verhalten zeigt. Kontinuierliche Prozesse. Die würde ich gerne mit Beispielen erklären. Wenn man Bierbrauen her nimmt, gibt es zwei Möglichkeiten. Die diskrete, also nicht-kontinuierliche Variante, wäre wenn man in einem geschlossenen Kessel die Zutaten hineingibt, zehn Liter Wasser zum Beispiel, und den Brauprozess einfach schrittweise ablaufen lässt. Am Ende kommt dann eine begrenzte Menge an Bier heraus. Die andere kontinuierliche Variante wäre, wenn man keinen vollkommen abgeschlossenen Kessel hat, sondern miteinander verbundene Kessel, bei denen immer wieder Zutaten zugefügt werden und immer wieder Bier entnommen wird. Das geht die ganze Zeit so, sodass man nicht nachvollziehen kann, welcher Liter Wasser zu welchem Liter Bier gehört. Dabei läuft ein Teilprozess im ersten Kessel ab während gleichzeitig im letzten Kessel der letzte Prozessschritt stattfindet bevor das Bier fertig wird. Und dann noch zum letzten Begriff, geschlossene Regelkreise. Ein geschlossener Regelkreis ist jene Logik in Form von Hardware oder Software, die das kontinuierliche Bierbrauen ermöglicht. Wenn man einen Prozess wie das kontinuierliche Bierbrauen hat, muss man schauen, wie man schlechtes Bier vermeidet während der Prozess läuft. Man möchte die Qualität auf einem gewissen Punkt halten. Beim schrittweisen Bierbrauen hat man nur die zehn Liter, bei denen etwas schief gehen kann. Und mit den nächsten zehn Litern macht man es dann besser. Aber was ist, wenn die Brauanlage dauernd läuft und ständig Bier austritt? Dann muss man währenddessen den Prozess überprüfen und schauen, dass man die gute Qualität des Bieres erhält. Das heißt, man testet oder misst Werte, die die Qualität beschreiben, überprüft, wie sich diese Werte von optimalen Werten unterscheiden, und reagiert entsprechend. Stimmt etwas beispielsweise mit dem Zucker- oder Alkoholgehalt nicht, muss das Mischverhältnis geändert werden. Das heißt, in einem geschlossenen Regelkreis werden während der Prozess läuft, gewisse Werte überprüft. Diese werden mit optimalen Werten verglichen und je nach Abweichung reagiert das System darauf. So. Aus Informatiksicht bestehen kontinuierliche Prozesse aus einer sich ständig wiederholenden Abfolge von Zustandsabfragen und Regulierungen. Zustandsabfragen und Regulierungen sind jeweils traditionelle Code-Stücke, die sich auf Sensoren oder Aktoren beziehen. Um solche kontinuierlichen Prozesse konsistent formal zu beschreiben, zu modellieren und in weiterer Folge ausführen zu können, haben wir folgende Merkmale identifiziert. Ich würde Ihnen die Merkmale gerne einfach mal vortragen und danach würde ich Sie bitten, diese zu bewerten, und zwar als wichtig oder unwichtig, und ich würde Sie dann auch bitten, eine Begründung dafür abzugeben. Und zwar haben wir einmal. Verschiedene Zustandsabfragen- und Regulierungskombinationen sind unabhängig und können parallel ablaufen. Zweite Eigenschaft. Regulierungen folgen immer auf Zustandsabfragen. Drittens. Die Dauer von jeder Zustandsabfrage- und Regulierungskombination ist beschränkt. Viertens. Wenn Zustandsabfragen gewisse Ergebnisse liefern, wird das System beendet. Fünftens. Bevor das System beendet wird, muss es in einen konsistenten Zustand gebracht werden. Und letzter Punkt wäre, das resultierende System soll für Menschen verständlich sein. Zu Punkt Eins. Würden Sie sagen, verschiedene Zustandsabfragen- und Regulierungskombinationen sind unabhängig und können parallel ablaufen, das ist wichtig oder unwichtig? Und warum?

#00:08:53-6#

---

- 98 6. B: Ich würde jetzt tatsächlich sagen, die sind alle wichtig. Das finde ich als wichtig, weil, klar, so  
99 ein System ist komplex und da kann man jetzt nicht nur an einer Stelle eine Zustandsabfrage  
100 machen, sondern sollte wahrscheinlich mehrere Parameter parallel überprüfen, mit/ Habe ich  
101 das so richtig verstanden? Ja. Genau. Da finde ich das schon wichtig, also, //I: Okay. // gibt ja  
102 viele Sachen, die man kontinuierlich überprüfen muss. #00:09:20-6#
- 103 7. I: Okay, super. Danke. Punkt Zwei. Regulierungen folgen immer auf Zustandsabfragen. Würden  
104 Sie sagen, das ist wichtig oder unwichtig? #00:09:30-8#
- 105 8. B: Ja, würde ich sagen, ist wieder wichtig. Weil eine Regulierung zu machen, ohne dass man den  
106 Zustand abgefragt hat, macht ja wenig Sinn, weil man damit ins Blinde regulieren würde. Also,  
107 man muss ja vorher wissen, ist jetzt eine Regulierung notwendig. Ich weiß nicht, ob Zweitens so  
108 zu verstehen ist, dass auf eine Zustandsabfrage immer eine Regulierung folgt oder dass  
109 Regulierungen nur folgen können auf eine Zustandsabfrage, also. Je nachdem, würde ich  
110 zustimmen oder nicht. (lacht) #00:10:04-0#
- 111 9. I: Eher zweiteres. #00:10:06-2#
- 112 10. B: Eher zweiteres? Also auf eine Zustandsabfrage muss immer eine Regulierung folgen?  
113 #00:10:09-8#
- 114 11. I: Also eine Regulierung/ Ach so. Dann habe ich Sie falsch verstanden. Eine Regulierung sollte nur  
115 geschehen, wenn eine Zustandsabfrage vorher // B: Ja. // erfolgt ist. Eigentlich das.
- 116 12. B: Sonst ist es ja eine Black Box. Was reguliert man da? Was passt man an, wenn man nicht  
117 vorher den Zustand abgefragt hat? Ja, genau. #00:10:27-4#
- 118 13. I: Okay. #00:10:27-6#
- 119 14. B: Drittens. Die Dauer ist beschränkt. Ja, das würde ich jetzt vielleicht als unwichtig einstufen,  
120 weil ich, ja, gut. Ist auch wichtig, weil man sollte es natürlich so schnell wie möglich durchführen,  
121 dass man so schnell wie möglich eingreifen kann. Aber, ja, wie lange das dann dauert, ob das so  
122 wichtig ist, kommt mir jetzt im Vergleich zu den anderen Punkten eher unwichtig vor, die Dauer.  
123 #00:11:02-3#
- 124 15. I: Punkt Vier. Wenn Zustandsabfragen gewisse Ergebnisse liefern, wird das System beendet.  
125 #00:11:12-7#
- 126 16. B: Ja, finde ich sehr wichtig, weil/ Klar, dafür macht man die Zustandsabfragen. Um zu  
127 überprüfen, ob alles okay ist und wenn irgendetwas jetzt super schiefläuft, bevor dann die  
128 Maschinen kaputt gehen oder irgendetwas, dann sollte man lieber das System beenden, also,  
129 bei gewissen Ergebnissen schon. #00:11:32-8#
- 130 17. I: Punkt Fünf. Bevor das System beendet wird, muss es in einen konsistenten Zustand gebracht  
131 werden. #00:11:40-6#
- 132 18. B: Ja. Ja, finde ich auch wichtig, weil man kann nicht einfach die Maschinen in jedem Zustand  
133 abschalten oder egal, was für ein Prozess es gerade ist, sondern muss schauen, dass, klar, es  
134 irgendwie in einem Zustand ist, in dem man es beenden kann. Also finde ich wichtig.  
135 #00:11:58-9#
- 136 19. I: Und der letzte Punkt. Das resultierende System soll für Menschen verständlich sein.  
137 #00:12:04-5#
- 138 20. B: Ja, vermutlich der wichtigste Punkt. Also, als Consultant hatte ich auch immer die Probleme,  
139 dass wenn die Anwender es nicht verstehen, dann lassen sie dieses (unv.) Systeme (unv.) Sachen,  
140 die man sich baut, meistens links liegen und verwenden das gar nicht richtig, vertrauen dem

- 141 Ganzen nicht, überprüfen es doch noch manuell. Also, ja. Das resultierende System sollte für die  
142 Menschen sehr verständlich sein. Das ist super wichtig. #00:12:29-9#
- 143 21. I: Nach diesen ganzen Punkten, zu Frage Zwei. Können Sie grafische Eigenschaften nennen, die  
144 Sie für die Modellierung kontinuierlicher Prozesse wichtig finden und ergeben sich daraus  
145 vielleicht Merkmale, die wir hier in dieser Liste von den sechs Eigenschaften vergessen haben?  
146 #00:12:51-0#
- 147 22. B: Nein, muss ich tatsächlich sagen, kann ich so adhoc nicht/ also da müsste ich mich  
148 wahrscheinlich mal hinsetzen mit einem kontinuierlichen Prozess und den versuchen, grafisch  
149 darzustellen. Dann vielleicht, aber/ #00:13:09-9#
- 150 23. I: Die Frage Drei könnte dann vielleicht auch ein bisschen adhoc gestellt sein. Und zwar, wenn Sie  
151 jetzt so darüber nachdenken, wie Sie einen kontinuierlichen Prozess modellieren würden, wo  
152 liegen da Ihrer Meinung nach die Herausforderungen? Oder wo könnten sie liegen?  
153 #00:13:37-0#
- 154 24. B: Ja, ich befürchte es ist die gleiche Antwort wie bei Zwei, also. Da bräuchte ich jetzt ein Beispiel  
155 um das durchzugehen so in meinem Kopf. So adhoc fällt mir das schwer. #00:13:54-2#
- 156 25. I: Dürfte ich Sie vielleicht fragen, wenn Sie an kontinuierliche Prozesse denken und Sie ja schon  
157 verschiedenste Prozesse modelliert haben, würde Ihnen da vielleicht selber ein Beispiel dafür  
158 einfallen, was ein kontinuierlicher Prozess sein könnte? #00:14:12-9#
- 159 26. B: Nein, gerade fallen mir eher diskrete Prozesse ein. (lacht) #00:14:23-5#
- 160 27. I: Okay, gut. Dann komme ich mal zur Einführung oder zur Vorstellung der Erweiterungen. Ich  
161 werde Ihnen Prozesse zeigen, die mit BPMN 2.0 und mit unseren Erweiterungen modelliert  
162 wurden. Die Erweiterungen sollen zum einen vordefinierte Modellierungskonventionen für in  
163 der Prozess- und Steuerungstechnik übliche Routinen bereitstellen, und zum anderen auch  
164 helfen die Unterschiede zwischen den parallelen Pfaden in den Prozessmodellen zu visualisieren.  
165 Die Prozesse werden in der [REDACTED] modelliert. Ist Ihnen  
166 diese Anwendung ein Begriff? #00:15:04-5#
- 167 28. B: Ja, ich habe es aber noch nicht selbst verwendet. #00:15:08-7#
- 168 29. I: Okay, gut. Aber wissen Sie grundsätzlich, was man damit ausführen könnte oder wie man sie  
169 grundsätzlich benutzen könnte, wie sie aufgebaut ist? #00:15:17-9#
- 170 30. B: Ich habe mal kurz einer Veranstaltung beigewohnt, wo sie verwendet wurde. Ja, die  
171 Grundzüge würde ich sagen. Ja. #00:15:26-3#
- 172 31. I: Okay, dann frage ich nochmal kurz nach, weil in unseren Prozessbeispielen kommen nämlich  
173 auch einige eigene Symbole vor, also eigene Extensions für die [REDACTED], die jetzt nicht Teil dieser  
174 Arbeit sind, sondern schon früher eingeführt wurden. Und zwar sind das Service Calls mit einem  
175 Zahnrad dargestellt, Scripts und Service Calls mit Scripts. Sind Ihnen diese Symbole, diese Tasks  
176 ein Begriff? #00:15:58-9#
- 177 32. B: Nein. #00:16:00-7#
- 178 33. I: Okay, gut. Bei Service Calls kann man im Grunde, also da die [REDACTED] hauptsächlich mit HTTP  
179 arbeitet, kann man über Service Calls im Grunde HTTP Requests absetzen. Bei Scripts kann man  
180 Code-Abschnitte in Ruby programmiert abarbeiten lassen, und eine Kombination aus beiden ist  
181 im Grunde Service Calls mit Script, wobei zuerst ein Service Call, als ein Get Request oder ein Put,  
182 Post, wie auch immer, abgesetzt werden kann. Und dann eventuell sogar mit einem Resultat  
183 daraus ein Script ausgeführt werden kann. Dass Sie da den Hintergrund dazu auch wissen. Zu  
184 den Erweiterungen im Zuge dieser Arbeit. Das Erste ist einmal das Closed Loop Subsystem

Gateway. Das Gateway ist eine Kombination aus einem inklusiven und einem ereignisbasierten Gateway. Es enthält Verzweigungen beziehungsweise Kanten, die für die Zustandsabfragen- und Regulierungsphasen des Zyklus ausgelöst werden, sowie Kanten, die beim Empfang von Abbruchereignissen ausgeführt werden. Die Ereignisse und Tasks in den einzelnen Kanten sind unabhängig voneinander. Damit erfüllen wir das erste der oben genannten Features, dass einzelne Verläufe unabhängig voneinander sind und sie parallel ausgeführt werden. Das Gateway ermöglicht außerdem die Definition der Intervalldauer jedes Zyklus sowie von Überschreitungsbedingungen im Sinne von Warten oder wait beziehungsweise Abbruch oder cancel, und der Ausführungsreihenfolge für Zustandsabfragen und Regulierungen. Beziehungweise könnte man auch sagen, Mess- und Steuerungsaufgaben. Man kann also die Attribute hier, Interval duration overrun mit cancel oder wait definieren. Beziehungweise Measure control cycle execution mit parallel oder sequentiell. Zur Erklärung, wait oder cancel. Wenn wait gewählt wird, beginnt die nächste Iteration wenn alle Verzweigungen beendet sind, alle Kanten beendet sind, und die festgelegte Intervalldauer erreicht ist. Bei cancel definiert die Intervalldauer genau die Zeit, in der jeder Zweig zu beenden ist. Wenn die Tasks in einem Zweig schneller beendet werden, wird der Zweig warten. Wenn noch nicht alle Tasks beendet sind, werden sie abgebrochen. Der Unterschied zwischen parallel und sequentiell. Bei parallel werden die Tasks nach Measure und Control Events parallel ausgeführt, bei sequentiell werden die Tasks nach Control Events erst ausgeführt, nachdem alle Tasks nach Measure Events beendet sind. Ja. Und dann gibt es noch Intermediate Catching Events, die hier eingeführt wurden. In einem Closed Loop Subsystem werden spezifische Ereignisse erwartet, die in eine der drei folgenden Kategorien fallen. Ereignisse für Zustandsabfragen beziehungsweise Messungen, Ereignisse für Regulierungen und Ereignisse für die Unterbrechung des Closed Loop Subsystems. Es gibt für jede Ereigniskategorie zumindest eine Kante, die vom Gateway ausgeht. Die Kanten zeigen an welche Tasks nebeneinander ablaufen. Sobald diese Ereignisse eintreten, werden auch die Tasks, die in den Kanten danach angeordnet sind, ausgeführt. Hier sehen jetzt gleich ein Bild eines Closed Loop Subsystems, in dem nur die Ereignissymbole inkludiert sind, die Ereignisse der drei Kategorien, ohne darauf folgende Tasks. Das heißt, wir haben hier einen Graphen wie man ihn in der  kennt. Haben das Closed Loop Subsystem hier eingefügt und es werden auch gleich drei Kanten erstellt, eine für ein Measure Events, eine für ein Control Event und eine für ein Cancel Event. Das heißt um diese drei Kategorien geht es. Und zwar. Measure Events. Empfängt Events für die Ausführung von Tasks in Messzyklen, beziehungsweise Zustandsabfragezyklen. Control, empfängt Events für die Ausführung von Tasks in Regelzyklen. Und Cancel empfängt Events für das Abbrechen von Closed-Loop-Systemen. Diese Symbole geben den Zweck der nachfolgenden Tasks an. Diese Tasks werden nur ausgeführt, wenn die Ereignisse ausgelöst werden. Das bedeutet, dass das Messereignis angibt, dass die nachfolgenden Symbole nur Messabläufe beziehungsweise Zustandsabfragen anzeigen. Das Gleiche gilt für Regulierungs- oder Kontroll- und Abbruchereignisse. Für Zustandsabfragen und Regulierungen können wir eine Zykluszeit definieren. Dadurch kann die Dauer von Anpassungen im System definiert werden. Je nachdem, ob das Closed Loop Subsystem einen parallelen oder sequentiellen, oder einen Wait- oder Cancel-Ansatz verfolgt, läuft die Ausführung unterschiedlich. Mit diesen Bedingungen kann man definieren, inwiefern Anpassungen beim System erfolgen. Hier sehen Sie ein Closed Loop Subsystem mit einem Task für eine Messung. In diesem Fall wird das Ereignis für die Messung alle zehn Sekunden getriggert. Danach wird der Wert V 1 geholt beziehungsweise gemessen. Wait bedeutet hier, dass ein neuer Zyklus erst startet, wenn die Messung erfolgt, das heißt der Prozess in dieser Kante abgeschlossen ist. Mit Cancel wird nach zehn Sekunden automatisch der neue Zyklus gestartet. Das heißt im Grunde haben wir hier einen Service Call nach dem Measure Event eingefügt. Und bei diesem Service Call wird ein Get Request an einen bestimmten Endpunkt abgesetzt. Und mit diesem Request wird der Wert für V 1 geholt. Für Measure Events kann man hier die folgenden Attribute definieren. Man kann die Interval frequency definieren, das heißt, Frequenz also in Hertz angegeben, Null Komma Eins, also einmal alle zehn Sekunden. Und man kann auch den Wert angeben, bei dem man erwartet, dass er sich ändert. Das heißt in unserem Fall V 1. Man kann aber auch, wenn man in der Measure-Kante mehrere Service Calls definieren möchte, die halt nacheinander erfolgen, auch hier verschiedene Werte angeben. Das heißt wenn zum Beispiel nicht nur V 1 abgefragt wird, sondern V 2, V 3, als Beispiel. Mithilfe von Regelungereignissen kann ferner festgelegt werden, welches Reglermodell verwendet wird, also PID, PI, PD-Regler zum Beispiel. Diese Regler werden in ihrer

242 mathematischen Form dargestellt. Die Tasks für sie sind im Grunde Berechnungen, die in festen  
243 Teilprozessen dargestellt werden. Nach diesen Berechnungen kann der Benutzer Tasks zu  
244 weiteren Datenverarbeitung hinzufügen. Dies kann zum Beispiel auch nach Mess-Tasks  
245 geschehen, und diese könnte man in diesem Zusammenhang auch einfach als  
246 Datenerfassungs-Tasks bezeichnen. Hier sehen Sie ein Prozessmodell mit einem Wert, der  
247 gemessen wird, und einer darauf folgenden Regelung. Das heißt, wir haben hier einmal das  
248 Measure Event, danach haben wir wieder unseren Service Call, der V 1 abholt, und dann haben  
249 wir in der Kante daneben mit dem Control Event einmal die Berechnung der Differenz vom  
250 optimalen Wert, also  $V_{opt}$  minus V 1, dann haben wir hier zum Beispiel in einem Script, also  
251 Script Differenzberechnung, Script PID Code, die Berechnung des PID-Reglers, also des  
252 Reglermodells, mit dem jeweiligen Wert, den wir heranziehen, um den Aktor anzusteuern in  
253 unserem Prozess. Oder entsprechend ein Kommando an das Element zu schicken, das aktiv auf  
254 den Prozess Einfluss nimmt. Das wäre hier auch wieder durch einen Service Call dargestellt. Und  
255 zwar Send new manipulated value. Wie Ihnen auffällt, haben wir hier aber noch nichts definiert  
256 für die Abbruchbedingung. Das kommt jetzt noch dann gleich. Wir können in der [REDACTED]  
257 Datenelemente definieren, die gewisse Werte haben als Default values, oder generell zur  
258 Berechnung der verschiedenen Reglermodelle. Und bei Control Event, was können wir hier  
259 definieren? Wie gesagt, auch die Zykluszeit, also die Intervalldauer, könnte man sagen, in Hertz  
260 angegeben. Und beziehungsweise kann man hier auch etwas genauer angeben, um welchen  
261 Regler es sich handelt. In unserem Fall haben wir hier einen PID-Regler angesetzt. Welcher Wert  
262 würde geändert werden? Das wäre in dem Fall MV, also Manipulating value. Und wir können  
263 hier auch einen maximalen und einen minimalen Wert sozusagen als Ranges angeben. Als  
264 Sicherheitsmaßnahme. Wait bedeutet wieder, dass für den nächsten Zyklus auf das Beenden  
265 aller Tasks gewartet wird, auch auf die Regulierungs-Tasks. Sequential heißt, dass die Tasks  
266 nacheinander ausgeführt werden würden. Das heißt, es wird erst gemessen beziehungsweise  
267 der Zustand abgefragt und mit diesem Wert, mit diesem gemessenen Wert, wird die Regelung  
268 dann durchgeführt. Dabei, wie gesagt, wird vom optimalen Wert,  $V_{opt}$ , der aktuelle Wert, V 1,  
269 abgezogen und mit dieser Differenz wird der neue Stellenwert MV mithilfe des PID-Modells  
270 ausgerechnet. Dieser wird dann in einem Service Request an das entsprechende Stellglied  
271 geschickt. #00:26:03-4#

272 34. B: Darf ich eine Frage stellen, oder? #00:26:07-5#

273 35. I: Ja, sicher. Ja. Entschuldigung, wenn ich so schnell bin, ja. #00:26:10-3#

274 36. B: (lacht) Nur eine Verständnisfrage. Also, ich hatte das im Studium so, dass/ Es wird ja nicht  
275 immer direkt eingegriffen, sobald es abweicht, ein bisschen vom  $V_{opt}$ , sondern halt erst, wenn  
276 es über eine bestimmte Schwelle ist. War das jetzt dargestellt praktisch, in diesem rechten  
277 Kasten da? Oder greift ihr immer sofort ein und passt an, sobald es vom  $V_{opt}$  abweicht, das  
278 Gemessene? #00:26:35-7#

279 37. I: Also, ja, einerseits soll die Schwelle dafür, also diese Range, dafür dienen, andererseits würde  
280 man dann hier auch im Script einen entsprechenden Abgleich dann einfügen. Also wir haben uns  
281 das offen gelassen, dass wir die Vorgabe für die Scripts eventuell auch noch anpassen können im  
282 jeweiligen Prozess. Oder dass wir vielleicht sogar noch einen zusätzlichen Schritt hier hinein  
283 modellieren und sagen, 'Okay, dieser Schwellwert darf nicht überschritten werden oder muss  
284 überschritten werden, je nachdem, wann der Regel eingreifen soll.' #00:27:10-7#

285 38. B: Okay. Ich dachte mir nur gerade, so welche drei Fälle können denn eintreten nach der  
286 Messung. Das eine haben wir dargestellt, Anpassung. Das andere wäre Abbruch, wenn es jetzt  
287 ganz schlimm ist oder der dritte wäre halt im Grunde nichts machen, weil es okay ist. Aber das  
288 hatte mir so ein bisschen gefehlt, aber okay. Dann kann man das ja hinein modellieren noch, ja,  
289 perfekt, das/ Okay. #00:27:32-7#

290 39. I: Das ist auch der Vorteil dabei, dass man das recht flexibel eigentlich halten kann. Ja. Würde  
291 hier parallel verwendet werden/ Also nur ganz kurz noch dazwischen. Wenn Sie Fragen haben,  
292 können Sie natürlich mich jederzeit unterbrechen. Ich wollte nur auch ein bisschen bei diesen

293 Sachen zügiger durchgehen, damit wir auch nicht zu lange mit dem Interview brauchen.  
294 #00:27:57-9#

295 40. B: Ja, passt. #00:27:59-5#

296 41. I: Damit das für Sie nicht ermüdend ist. Würde hier parallel verwendet werden, würde der letzte  
297 Wert von V 1 genommen werden, für den keine Zeitgarantie besteht. Das heißt, das wäre dann  
298 einfach der Wert aus dem letzten Zyklus, aus dem letzten Zeitfenster, könnte man sagen. Der  
299 letzte Wert, der auf die Variable geschrieben wurde. Zustandsabfragen und Regulierungen  
300 sollten in regelmäßiger Frequenz ausgelöst werden. Abbruchereignisse hingegen werden nur  
301 durch ihre Abbruchbedingungen ausgelöst, die der Benutzer definieren kann. Ein Beispiel für ein  
302 Abbruchereignis wäre, wenn etwas den Abbruch eines Zyklus auslöst, wie zum Beispiel  
303 Notstopp-Funktion. Notstopp, das wurde auch in einem anderen Interview angemerkt, ist  
304 vielleicht ein etwas sehr extremes Ereignis, aber man könnte sich hier natürlich auch etwas  
305 Anderes, was einfach den Reglervorgang unterbrechen würde, was einfach den kompletten  
306 Zyklus unterbrechen würde, vorstellen. Irgendein Kommando von außen beispielsweise. Also nur,  
307 dass Sie sich ein konkretes Bild davon machen, wie es funktionieren kann. Die Bedingung auf  
308 jeden Fall, die hierfür hergenommen wird, kann man hier im Label neben dem Abbruchereignis,  
309 neben dem Cancel-Ereignis, definieren. In unserem Beispiel wäre es jetzt zum/ beispielsweise  
310 Emergency Stop Active müsste auf true gesetzt werden. Das heißt, sobald der Notstopp aktiviert  
311 wurde, würde aus dem Zyklus ausgebrochen werden und das System würde beendet werden.  
312 Also das Modell wäre damit auch beendet, könnte man sagen. Als Default würden wir hier aber  
313 natürlich davon ausgehen, dass die Bedingung false ist. Weil sobald die natürlich ausgelöst wird,  
314 müsste man gar nicht erst den Prozess weiterführen. (...) Und hier sehen Sie jetzt einen Prozess,  
315 bei dem Aufräum-Tasks definiert wurden. Es gibt natürlich die Möglichkeit, dass wir nach Cancel  
316 Events auch noch Tasks hinein modellieren, die das System in einen konsistenten Zustand, wie  
317 vorhin schon erwähnt, überführen können. Das heißt, wenn noch gewisse Bedingungen erfüllt  
318 werden müssen oder noch gewisse Tasks ausgeführt werden müssen bevor man das System  
319 beendet. In dem Fall könnte man sagen, 'Okay, Initiate shutdown routine für den ersten Kessel,  
320 also, for vessel 1.' Oder dergleichen. Also hier mal dargestellt als Service Call. Also, ein  
321 Kommando, das nach außen abgesetzt wird. Ja. Die vorgestellten Erweiterungen sollen bei der  
322 Modellierung von kontinuierlichen Prozessen helfen, indem Vorlagen für die Erstellung von  
323 Prozessmodellen vorgegeben werden. Und andererseits durch die Darstellung als Closed Loop  
324 Subsystem mit eigenen Symbolen für Zustandsabfrage-, Regulierungs- und Abbruchereignisse  
325 helfen, solche Prozesse leichter nachvollziehen zu können. Hinzu kommt, dass man für eine  
326 übersichtlichere Darstellung des gesamten Prozesses auch Subprozesse zur Unterteilung nutzen  
327 kann. Damit erfüllen wir auch das letzte Feature, dass wir zur Verständlichkeit der Modelle  
328 beitragen wollen. Okay. Beispiele für Prozessmodelle. Ich werde Ihnen nun Prozessbeispiele  
329 zeigen, die mit den in unserer Arbeit vorgestellten Erweiterungen modelliert sind. Ich möchte,  
330 dass Sie sich die Modelle ansehen und mir sagen, was Sie aus ihnen herauslesen können, und ob  
331 die Modelle den notwendigen Informationsgehalt für die Modellierung der zugrundeliegenden  
332 Regelungsprozesse erfüllen. Vorab wird Ihnen zum jeweiligen Prozess erklärt, was abgebildet  
333 werden soll. Und ich würde Sie bitten, einfach offen Feedback dazu zu geben. In beiden Fällen  
334 wird im Grunde ein Heizprozess dargestellt. Im ersten Fall haben wir hierfür einen  
335 Wärmetauscher, der mit einem Dampfstrom versorgt wird und damit einen Rührkessel erwärmt.  
336 Und im zweiten Fall wäre es ein ganz einfaches Heizelement. Zum ersten Beispiel mit einem  
337 einfachen PI-Regler. Temperaturregelung mit einem Wärmetauscher, wie gesagt, basierend auf  
338 einem Beispiel aus der MathWorks-Bibliothek. Die Temperatur einer Flüssigkeit in einem  
339 Rührkessel wird mittels Wärmetauscher geregelt. Der über den Wärmetauscher eingebrachte  
340 Wärmestrom wird über ein Ventil, das den Dampfstrom kontrolliert, gesteuert. Der zu  
341 beachtende störende Umgebungseinfluss ist in diesem Fall die schwankende Temperatur der  
342 zugeführten Flüssigkeit. Der Tank hingegen, also die Tankwand, gilt aber als isoliert. Das heißt,  
343 wir gehen nicht davon aus, dass wir über Wärmeübertragung an die Umgebung irgendwelche  
344 Wärmeverluste hätten. Das Flowchart für den Prozess würde so aussehen, also ein einfacher,  
345 wie gesagt, ein einfacher Rührkessel mit einem Wärmetauscher, Zufuhr einer zusätzlichen oder  
346 der einfließenden Flüssigkeit über das Rohr darüber, ein entsprechendes  
347 Temperaturmesselement, und hier haben wir die Steuerung des Ventils. Das heißt, wir hätten

mal einige Datenelemente definiert, die eventuell auch später für die Berechnung des PI-Modells wichtig sein könnten, aber auch sonstige Defaultwerte oder Variablen, die im Zuge des Prozesses oder im Verlauf des Prozesses überschrieben werden. Was jetzt eigentlich/ Als Beispiel auch noch ein paar Endpunkte, aber wenn Sie grundsätzlich die [REDACTED] schon mal gesehen haben, wissen Sie, dass man hier auch Endpunkte definieren kann, für die Service Calls oder dergleichen. So. Was wir jetzt hier haben, ist einmal Measure von der Temperatur des Tanks. Einmal Measure für die Temperatur der Störung, also des Zuflusses. Wir haben einmal Control. Wir haben nach dem Control Event das Modell für den PI-Regler. Wir haben dann eventuell noch eine Conversion, wenn wir das möchten, also als Beispiel, wenn wir noch eine Umrechnung brauchen sollten. Und sobald wir den Wert dann in der entsprechenden Größe haben oder in der entsprechenden Form haben und ihn an den Aktor schicken können, das heißt in dem Fall eigentlich den Antrieb für das Ventil, dann können wir den Wert auch über einen Service Call nach außen schicken. Unsere Bedingung für das Beenden des Modells oder des Prozesses gab es im Modell der MathWorks-Bibliothek nicht vorgegeben. Deswegen haben wir hier generell einfach mal Stop activated, sobald das true gesetzt wird, initiieren wir hier als Script dargestellt, aber könnte auch Service Call mit Script oder einfach nur Service Call sein, Execute shutdown sequence. Das heißt das könnte zum Beispiel, wenn man das zur besseren Übersichtlichkeit gestalten möchte, auch einfach ein Subprozess sein, der aufgerufen wird. Und nachdem das aufgerufen wurde, gehen wir aus dem Closed Loop Subsystem heraus und der Prozess würde beendet werden. Hier sehen Sie noch wie das Script ausschauen würde, dass wird das mathematische Modell hätten für den PI-Regler. Hier ein Beispiel, was man zum Beispiel noch hineinschreiben könnte für die Umwandlung. In dem Fall ist es nur ein Überschreiben einer Variablen auf eine andere. Es soll einfach nur verdeutlichen, dass man hier in der Modellierung wie ich es vorhin schon erwähnt habe recht frei ist und dass man noch unterschiedliche Tasks in einer Reihe nacheinander modellieren kann wenn man das möchte. Ja, und zur Bewertung des Modells haben wir ein paar Kriterien. Nämlich Verständlichkeit. Also ist das für Sie ersichtlich, was grundsätzlich passiert? Übersichtlichkeit. Können Sie das Gesamtsystem auf einen Blick erfassen in diesem Modell? Einfachheit. Könnte man das Modell noch einfacher darstellen Ihrer Meinung nach? Logik. Wird klar, was parallel und was sequentiell passiert? Und schließlich Erweiterbarkeit. Könnte man dem Modell noch etwas hinzufügen, was den Informationsgehalt verbessern würde? Und diese Kriterien, würde ich Sie bitten auf einer Skala von Eins bis Fünf zu setzen, wobei Eins die schlechteste Bewertung wäre, also das heißt je mehr Punkte das Modell kriegt, desto besser. Also. Eins, sehr schlecht, und bis Fünf, sehr gut. #00:37:05-4#

42. B: Ja, es geht jetzt nur um diesen Closed Regelkreis Loop, also es geht jetzt nicht darum, irgendwie komplett darzustellen, was passiert? Wie beim Flowchart auch, also? #00:37:17-4#

43. I: Es geht im/ Also es geht nicht um das physikalische System an sich, es geht eher so um die Prozesslogik könnte man sagen. #00:37:27-2#

44. B: Eh, klar. #00:37:31-7#

45. I: Also, es ist jetzt zum Beispiel/ Natürlich haben wir hier jetzt kein, wie in einem CAD File, haben wir hier natürlich nicht den Umfang des Kessels, wir haben keine Rahmenbedingungen für das physikalische Modell, also, im Sinne von, wie verhält sich zum Beispiel das Fluid im Rührkessel? Wir gehen davon aus, dass wir hier eine ideale Durchmischung haben. Solche Sachen, die natürlich dann eher in FEM, in CFD-Simulationen einfließen würden. Also ich nehme an, das sind Ihnen eh Begriffe, die Sie vielleicht kennen. Das wäre dann natürlich um einiges umfangreicher und in unserem Sinne wollen wir eigentlich nur die Möglichkeit haben, dass wir auch/ dass wir die grundsätzlichen Abläufe dahinter abbilden können. Also physikalische Bedingungen und dergleichen eigentlich nicht. #00:38:26-3#

46. B: Okay. Aber man // I: Ich hoffe, das veranschaulicht/ // sollte schon jetzt anhand von dieser Darstellung verstehen,'Okay, da wird Temperatur gemessen und da kommt ein Wärmetauscher hinein, und alles.' Das sollte schon hier heraus (unv.)? // I: Ja/ // #00:38:40-0#



- 398 47. I: Aber ich meine, dass es ein Wärmetauscher konkret ist, eher nicht, aber dass es einen Wert  
399 gibt, der geregelt wird, und in dem Fall ist es halt ein Temperaturwert, aber das müsste man  
400 dann aus vielleicht aus der Bezeichnung der Variablen herauslesen können. Ja. Also es geht eher  
401 um die generische Darstellung solcher Prozesse, könnte man sagen. (...) Ja. #00:39:09-7#
- 402 48. B: Okay. #00:39:12-0#
- 403 49. I: Das erste Kriterium wäre, wenn Sie das eh gemeint haben, Verständlichkeit. Ist für Sie  
404 grundsätzlich ersichtlich, was hier vom Prozessablauf passiert? #00:39:25-0#
- 405 50. B: Also, der Mess- und Regelprozess wird mir daraus sehr klar, ja. Das würde ich mit einer Fünf  
406 bewerten. Verständlichkeit. Vom Gesamtprozess jetzt, wirklich was wird damit wo erhitzt und so.  
407 Also das ganze Big Picture, da finde ich es visuell noch nicht so gut. Also da fand ich die  
408 Darstellung, was war das andere? Die //I: Flowchart. // Prozessdarstellung, die Notation. Die  
409 noch zusätzlich zu haben, finde ich dann schon hilfreich, also wenn man das Big Picture  
410 verstehen will. #00:39:58-3#
- 411 51. I: Okay. #00:40:03-3#
- 412 52. B: Also dahin ging auch so ein bisschen meine Frage. Deshalb wusste ich nicht/ Jetzt nur in Bezug  
413 auf diesen Regelkreis, dann finde ich es sehr gut dargestellt. #00:40:11-9#
- 414 53. I: Es geht hier ein bisschen wie in der Introduction angesprochen eher darum, dass wir mal uns  
415 grundsätzlich anschauen, okay, wo ist jetzt der Unterschied zwischen diskreten  
416 Fertigungsprozessen, wo man grundsätzlich sagen könnte, 'Okay, ich habe hier jetzt ein  
417 Werkobjekt oder ein Werkstück und das kommt hier mal in diese eine Maschine hinein und wir  
418 haben einmal einen Fräsprozess, wir haben einmal eine Bohrung, die gesetzt werden muss, oder  
419 dergleichen. Vielleicht sogar wird automatisch ein Gewinde geschnitten oder so etwas. Und  
420 dann sind diese Prozessschritte aber bei einem gewissen Punkt erledigt und dann würde es  
421 wieder in den nächsten Prozessabschnitt wandern. Und irgendwann ist das Teil, das eine Objekt,  
422 vielleicht fertig oder eine Charge davon ist fertig. Und natürlich verhalten sich dann  
423 kontinuierliche Prozesse bis zu einem gewissen Grad einfach anders, und das versuchen wir hier  
424 so ein bisschen hervorzuheben oder heraus zu arbeiten, könnte man sagen. Ja. Zweiter Punkt  
425 wäre Übersichtlichkeit. Was würden Sie sagen? #00:41:26-5#
- 426 54. B: Ja, das ist auch wieder für mich also das Reglersystem sehr, sehr übersichtlich. Gesamtsystem.  
427 Ich weiß nicht. Also irgendwie fehlt mir da so der Anfang und das Ende vom Gesamtsystem, also  
428 diese Übersicht, die man von der anderen Darstellung hat. Die Übersichtlichkeit von der  
429 Überprüfung finde ich sehr gut, aber ist für mich halt eben ein Subprozess, nicht das  
430 Gesamtsystem. #00:41:57-7#
- 431 55. I: Was würden Sie grundsätzlich als/ Was würde Ihnen fehlen, um das Gesamtsystem abzubilden?  
432 Also was fehlt Ihnen da an Informationen, konkret? #00:42:08-6#
- 433 56. B: Ja, wie Sie schon sagten. So ein bisschen, dass die physikalischen Gegebenheiten, was für eine  
434 Flüssigkeit, wie oft kommt da neue Flüssigkeit hinein. Solche Sachen. Also allgemein. Wie ist das  
435 Ganze aufgebaut? Weil die Darstellung, wie wir sie jetzt bewerten, ist ja wirklich nur auf dieses  
436 Überprüfen eigentlich ausgerichtet. Also. Ja, irgendwie so das allgemeine Setup. Ich glaube, das  
437 fehlt mir so ein bisschen. // I: Okay, verstehe. // Ich bin auch immer ein Mensch fürs  
438 ganzheitliche Betrachten. Deshalb. Dann würde ich da vielleicht eher auf Gut gehen, ja.  
439 #00:42:53-6#
- 440 57. I: Okay, danke. Was würden Sie zur Einfachheit sagen? Könnte man Ihrer Meinung nach das  
441 Modell noch einfacher darstellen? #00:43:02-9#
-

- 442 58. B: Nein, ich denke nicht. Also, das ist ja schon sehr schön mit den Swimlanes und den kleinen  
443 Visualisierungen praktisch, also die einzelnen (lacht) Abschnitte, finde ich sehr gut, ja.  
444 Darstellung, Einfachheit. #00:43:21-2#
- 445 59. I: Dann kommen wir zur Logik. Wird für Sie klar, was parallel und was sequentiell passiert?  
446 #00:43:28-1#
- 447 60. B: Ja, können wir da nochmal kurz oben auf das Modell. Ich glaube schon. Danke. (...) Ja, also im  
448 Grunde passieren ja/ die vier Stränge können parallel passieren, aber natürlich muss erst einmal/  
449 Nein, im Grunde nicht. Nein, muss ich leider sagen. Nein. Weil hier ist es jetzt so dargestellt, als  
450 ob die vier Sachen parallel passieren, aber es muss natürlich erst einmal gemessen werden,  
451 bevor dann geregelt werden kann. Da bin ich auch so ein bisschen gerade/ Das ist für mich nicht  
452 ganz klar. #00:44:11-8#
- 453 61. I: Okay, das heißt, da war ich vielleicht auch ein bisschen zu schnell bei der Erklärung, was  
454 sequential in diesem Zusammenhang bedeutet. Also, der Marker alleine beim Closed Loop  
455 Subsystem mit wait und sequential, würden Sie sagen, dass Sie beschreiben könnten, was wait  
456 und sequential bedeuten nach meiner Einführung? Und würden Sie sagen, dass es eventuell  
457 einfach zu schnell war oder dass es doch etwas zu komplex war? #00:44:44-1#
- 458 62. B: Ich würde sagen, dass wir vielleicht bei der Einführung schon ein Beispiel von so einem  
459 Prozess durchgehen. Vielleicht würde das dem Verständnis noch helfen, ja. Also ich könnte jetzt  
460 nicht die Definition wiederholen (unv.). #00:44:59-4#
- 461 63. I: Also wait, warten, würde in diesem Zusammenhang, weil/ wir haben von MathWorks natürlich  
462 keine Vorgabe, wie jetzt so etwas in der Steuerung wirklich umgesetzt werden würde. Wir haben  
463 eigentlich nur das Reglermodell vorgegeben gehabt. Also ein Blockdiagramm könnte man sagen.  
464 Wir sind jetzt einfach davon ausgegangen, dass das System jeden Strang fertig werden lässt und  
465 dann erst in den neuen Zyklus übergeht. Aber wir haben auch sequential angegeben. Das heißt,  
466 dass zuerst sämtliche Tasks auf jeden Fall in den Messsträngen oder den Messkanten fertig  
467 werden sollten, und DANN erst Control ausgeführt werden kann. Das Gegenstück dazu wäre  
468 parallel. Bei parallel möchten wir dem Modellierer die Möglichkeit geben, dass er auch sagen  
469 kann, okay, grundsätzlich können die Measure- und die Control-Ereignisse gleichzeitig  
470 ausgewertet und ausgelöst werden. Das hieße aber dann natürlich, wenn jetzt Control aus  
471 irgendeinem Grund dann schneller fertig werden würde, könnte es nicht mehr mit dem letzten  
472 aktuellen Wert von Measure arbeiten. Also es müsste natürlich mit dem Wert, der vorhanden ist,  
473 arbeiten, aber dann wäre es nicht mehr aus dem gleichen Zyklus oder mit dieser Zeitgarantie.  
474 Wir hätten zwar noch ein gewisses Fenster, aus dem der Wert herausgenommen werden kann,  
475 aber wir haben nicht mehr die Garantie, dass wirklich zuerst in diesem Zyklus Measure Events  
476 fertig geworden sind, oder die Tasks nach den Measure Events abgearbeitet wurden, und DANN  
477 erst Control gestartet ist. Diese zwei Möglichkeiten gibt es. #00:46:42-0#
- 478 64. B: Okay, aber bei dieser Darstellung, da sieht das jetzt wirklich so aus, als ob die parallel ablaufen  
479 würden. Okay, wait, sequential, okay. Aber warum macht man das dann nicht so, dass die nie  
480 hier zum Beispiel aus dem Measure heraus erst in den nächsten Strom geht? Also, dass man  
481 sieht, das ist wirklich davon abhängig? #00:47:06-1#
- 482 65. I: Würde es vielleicht helfen eventuell, um zu sehen dass Measure gleichzeitig passieren können,  
483 also die beiden, die hier nebeneinander stehen, dass man die auf die gleiche Höhe hebt und  
484 dann zum Beispiel erst das Control kommt? #00:47:27-8#
- 485 66. B: Ja. #00:47:31-0#
- 486 67. I: Wissen Sie, was ich meine? Dass man das einfach auf die gleiche Höhe, ungefähr hierher setzt.  
487 #00:47:36-4#
-

- 488 68. B: Ja, das würde schon einmal helfen, weil das die beiden Measures sind. Genau, und dann wenn  
489 die beiden, dass dann erst das andere, also/ Ja, es ist/ Für mich würde es mehr Sinn machen,  
490 wenn die beiden auf der gleichen Höhe sind und parallel laufen können. Und dann praktisch  
491 unter ihnen das weitergeht mit dem Regler. Weil der Regler greift ja erst ein, wenn das andere/  
492 wobei da ist/ es läuft ja dann schon erst danach ab und nicht in einem parallelen Leitfaden, so  
493 wie es hier jetzt praktisch aussieht, für mich, auf den ersten Blick, ohne dass ich tief in der  
494 Materie bin. (lacht) Ist das verständlich? #00:48:13-3#
- 495 69. I: Ja, schon. Aber würde dadurch, dass wir jetzt beide Ereignisse oder beide  
496 Modellierungsvarianten eigentlich offen gehalten haben, also sequential und parallel, würden  
497 Sie sagen, dass man parallel komplett streichen könnte und dass Ihnen überhaupt keine  
498 Anwendung einfallen könnte, wo/ Ich will es jetzt nicht zu kompliziert formulieren. Parallel  
499 hieße ja, dass die Measure Events und die Control Events gleichzeitig beziehungsweise parallel  
500 ausgewertet werden, und damit gleichzeitig ausgelöst werden könnten. Das heißt die Stränge  
501 würden wirklich komplett parallel durchlaufen. Das hieße aber, dass das jetzige Control-Ereignis  
502 oder die Tasks, die auf das jetzige Control-Ereignis folgen, nicht unbedingt auf die aktuellsten  
503 Messwerte zugreifen, sondern einfach auf die zuletzt verfügbaren. //B: Also/ // Würde Ihnen/  
504 Entschuldigung. Sagen Sie nur. #00:49:29-0#
- 505 70. B: Wir hätten ja am Anfang die sechs Punkte, glaube ich, waren es, was so ein System erfüllen  
506 muss. Und da hatte ich ja auch gesagt, also, ich würde nie regulieren wollen, ohne dass ich  
507 vorher gemessen habe. Deshalb fällt mir jetzt eigentlich kein Beispiel ein, wo ich sie parallel sehe.  
508 Ich würde das immer da darunter sehen, sequentiell, aber, ja. Vielleicht wenn Sie da mehr  
509 verschiedene Prozesse modelliert haben, wo das vielleicht notwendig ist, das auch parallel zu  
510 tun, dann klar, kann man das in einem Extra-Strang lassen. #00:49:58-7#
- 511 71. I: Das ist jetzt auch ein bisschen die Frage. Also, natürlich sollten sie alle/ Also es sollte eine  
512 Regulierung wirklich nur erfolgen, wenn man den aktuellen Zustand des Systems kennt. Aber  
513 hier geht es eher darum, welchen Wert man dann wirklich nimmt. // B: Den älteren schon, okay.  
514 // Dass das vielleicht dann ein bisschen Sinn ergibt. #00:50:18-2#
- 515 72. B: Okay. #00:50:20-3#
- 516 73. I: Also wir sind da auch ein bisschen teilweise auf den Input von den Interviewgästen angewiesen.  
517 Das wir einfach auch vielleicht andere Sichtweisen hier hineinbringen können, und ein bisschen  
518 erfahren könnten, ob vielleicht jemandem von den Interviewgästen etwas einfällt, wofür man  
519 das dann hier auch einsetzen könnte. Aber natürlich also/ Hm? (nachfragend) Ja? #00:50:46-2#
- 520 74. B: Aber wenn man jetzt sagt, das ist sequentiell, das würde ja auch noch passen bei dem Beispiel,  
521 was Sie gerade gesagt haben. Man berechnet das, dann passt man es an, und gleichzeitig  
522 berechnet das schon das Neue. Wir passen das jetzt an auf Grundlage der alten Messung. Aber  
523 es kann ja trotzdem schon wieder neu hineingehen, oder? Oder kann man nicht wieder neu  
524 praktisch in die Loop hineingehen, während man noch reguliert? Wenn das verständlich ist.  
525 (lacht) #00:51:15-8#
- 526 75. I: Ja, ich sage es mal so. Wenn ich jetzt ein technisches Setup hernehmen würde, wie eine SPS/  
527 Ich weiß nicht, ob Ihnen bekannt ist, wie SPS'en grundsätzlich funktionieren. #00:51:31-3#
- 528 76. B: Nein. #00:51:32-7#
- 529 77. I: Okay, also man würde grundsätzlich mal ein Prozessabbild machen. Das heißt man würde  
530 sämtliche Inputs einfach mal speichern und diese Werte fixieren und mit diesen Werten dann  
531 auch weiter arbeiten. Also man hätte einfach mal quasi ein Register ausgefüllt mit den Werten,  
532 mit denen man dann eigentlich in die Berechnung des SPS-Programmes hineingeht. Dann wird  
533 wie gesagt das SPS-Programm mit einem Regler eventuell sogar ausgeführt. Das heißt hier wird  
534 wirklich der aktuelle Wert genommen, der wird verarbeitet und dann geht es erst wieder Eins zu  
535 Eins in die Outputs hinaus. Und das passiert alles in dieser einen Zykluszeit. Das heißt das

536 passiert auch alles unter Zeitgarantie. Jetzt ist es halt eine technische Anwendung. Aber wir  
537 wollten diese Möglichkeit auch nicht verbauen, weil wir gesagt haben, okay, vielleicht gibt es  
538 aber auch andere Anwendungsbeispiele, wo man sagen könnte, 'Okay ich muss jetzt zwar den  
539 Zustand eines Systems vielleicht überprüfen, aber eventuell kann es passieren, dass trotzdem  
540 eine Regulierung schon parallel überprüft werden kann oder los gestartet werden kann. Und  
541 dann kriege ich aber einen anderen Wert heraus. Oder ich kriege einen anderen Wert, mit dem  
542 ich dann arbeiten muss. Ist das //B: Okay, also die Regulierung in bestimmten Fällen nochmal  
543 anstoßen will, bevor man nochmal neu gemessen hat? // verständlich? Vielleicht, ja.  
544 #00:53:10-6#

545 78. B: Okay. Ja. #00:53:16-4#

546 79. I: Aber natürlich, die meisten Fälle wären sequentiell. Wenn man gerade technische Prozesse  
547 sich anschauen würde. #00:53:23-1#

548 80. B: Okay. #00:53:25-5#

549 81. I: Okay, das heißt, die Logik war für Sie nicht unbedingt ersichtlich? #00:53:34-5#

550 82. B: Ja, war für mich ein bisschen schwierig. Ja. #00:53:38-4#

551 83. I: Was würden Sie dann sagen? #00:53:40-5#

552 84. B: Drei. #00:53:46-4#

553 85. I: Okay. Und der letzte Punkt wäre Erweiterbarkeit. Könnte man im Modell noch etwas  
554 hinzufügen, was den Informationsgehalt verbessern würde? Das haben Sie auch schon vorhin ein  
555 bisschen angesprochen, dass da eventuell noch etwas hinzugefügt werden könnte. #00:54:01-7#

556 86. B: Ja, das geht für mich so ein bisschen in die ersten beiden Fragen. (Unv.) Also, ja, für mich so  
557 ein bisschen das Big Picture, das Physikalische, dass man eher den Aufbau noch so ein bisschen  
558 versteht, aber das ist vielleicht auch eine persönliche Präferenz. #00:54:16-2#

559 87. I: Okay, passt. Gut. Das zweite Prozessmodell ist im Grunde wie gesagt auch eine  
560 Temperaturregelung. Und zwar aus den Schulungsunterlagen der Firma Siemens entnommen. Es  
561 handelt sich hierbei ebenfalls um, wie gesagt, um eine Temperaturregelung für einen  
562 Rührreaktor, einen Rührkessel. Die Regelung wird in diesem Beispiel mit einem PID-Regler, also  
563 im Grunde ein anderes mathematisches Modell, abgebildet, inklusive einer Handsteuerung  
564 sowie einem Pulsgenerator, also Pulsweitenmodulation ist auch Teil des Modells. Die Heizung  
565 erfolgt nicht über einen Wärmetauscher, sondern wie gesagt über ein Heizelement. Und wir  
566 haben auch Verriegelungsbedingungen definiert. Also wir haben hier eine Bedingung für ein  
567 Mindestfüllniveau für den Reaktor. Das sind 200 Milliliter beziehungsweise haben wir auch eine  
568 Maximaltemperatur angegeben, also sechzig Grad. Nur um zwei Beispiele mal zu nennen. Als  
569 Basis für die Prozessmodellierung wurden Unterlagen für die Prozessmodellierung mit Simatic  
570 PCS 7 herangezogen. Unser Prozessmodell wird mit einer automatischen Steuerung modelliert,  
571 die mit Umschalten auf Handsteuerung aus dem Closed Loop System ausbrechen würde. Wir  
572 gehen davon aus, dass das System bereits angelaufen ist und automatisch gesteuert wird. Das  
573 heißt Hochfahren und dergleichen ist schon erledigt, wir sind im kontinuierlichen Zustand  
574 könnte man sagen. Und weiters wird der Prozess nur für einen Reaktor und nicht für zwei wie in  
575 den Unterlagen eigentlich beschrieben dargestellt. Wir haben wieder verschiedene  
576 Datenelemente eingefügt, wie gesagt auch die maximale Temperatur mit sechzig Grad. Das  
577 Mindestfüllniveau mit 200 Milliliter. Wieder verschiedene Parameter für die Berechnung des  
578 PID-Reglers. Und zum Beispiel in dem Fall auch Operation Mode, dass der auf Automatik  
579 gesetzt ist, und zum Beispiel auch den Hauptschalter, dass der auf On steht. Wir haben in diesem  
580 Fall cancel und sequential. Cancel, nochmal kurz zur Wiederholung, würde bedeuten, dass wir  
581 uns an die Zeitbedingungen halten müssen, also wir eine gewisse Zykluszeit vorgegeben haben  
582 und wenn diese Zykluszeit überschritten wird, wird alles beendet und in den nächsten Zyklus

583 gewechselt. Also es fängt dann schon der nächste Zyklus an, wenn wir eine gewisse Dauer  
584 überschreiten. So. Messungen, also was hätten wir für Zustandsabfragen, für Measure Events?  
585 Wir haben einmal natürlich Temperatur des Reaktors. Wir haben den Füllstand im Reaktor, also  
586 ich hoffe das ist für Sie eh gut ersichtlich. Das können Sie eh erkennen. Wir haben einmal zum  
587 Beispiel die Abfrage, welchen Operation Mode haben wir aktuell. Wir haben die Abfrage, ob ein  
588 Emergency Stop besteht. Das könnte aber auch als Push-Nachricht eventuell implementiert  
589 werden. Das war auch ein guter Input von anderen Interviews. Wir haben die Abfrage des  
590 Hauptschalters. Und dann haben wir schließlich hier ein Control wieder, ein Control Event mit  
591 PID-Controller, Pulsweitenmodulation und dann wird entsprechend das Signal an das jeweilige  
592 Stellglied ausgeschickt. Wir haben dann noch mehrere Abbruchbedingungen in den Unterlagen  
593 schon vorgegeben, also wie gesagt, die Verriegelungsbedingungen. Zum Beispiel Hauptschalter.  
594 Wenn der auf Off geschaltet wird, Emergency Stop wieder als Beispiel, ist aber natürlich ein  
595 etwas extremeres Beispiel dafür gegeben. Und dann etwas Interessantes, speziell hier für diesen  
596 Prozess. Die Temperatur des Reaktors, wenn die die Maximaltemperatur des Reaktors  
597 überschreitet beziehungsweise das aktuelle Füllniveau, wenn das unter das Minimalfülllevel  
598 kommt. Und dann haben wir noch Operation Mode, wenn der auf Manual gesetzt wird, dann  
599 rufen wir zum Beispiel hier Service Call mit Script auf oder einen Subprozess oder dergleichen. Ja,  
600 wieder mathematisches Modell für den Regler. Und ich würde Sie wieder wie vorhin bitten, das  
601 Modell zu bewerten, also wieder Verständlichkeit, Übersichtlichkeit, Einfachheit, Logik und  
602 Erweiterbarkeit. Kurze Zwischenfrage nur. Möchten Sie hier kurz eine Pause machen oder geht  
603 es für Sie noch? #00:59:14-8#

604 88. B: Ich glaube, kurze Pause wäre ganz gut für meinen/ #00:59:18-6#

605 89. I: Okay, einen Moment und/ dann stoppe ich hier kurz die Aufnahme. #00:59:24-6#

606 UNTERBRECHUNG - 5 Min. PAUSE

607 90. I: Okay, Aufnahme läuft wieder. Wir waren beim zweiten Prozessmodell und schon wieder bei  
608 der Bewertung. Beginnen wir mit Verständlichkeit. #00:00:17-4#

609 91. B: (Unv.) Ja, im Grunde wieder ähnlich wie bei dem Vorhergegangenen. Ich fand aber dieses hier  
610 für mich/ da hatte ich irgendwie weniger physikalische Fragen, vielleicht weil ich die Darstellung  
611 nicht gesehen hatte. Also hier mit dem Reaktorfüllstand und maximalen Temperatur finde ich  
612 eigentlich schon sehr verständlich in der Darstellung. Also da fehlt mir jetzt nicht groß das Big  
613 Picture, dass ich da hier physikalische Sachen bräuchte. Vielleicht weil ich mich jetzt auch schon  
614 mehr an die Darstellung gewöhnt hatte, aber ich würde dann, Verständlichkeit habe ich  
615 eigentlich keine offenen Fragen, sehr gut sagen. #00:00:55-4#

616 92. I: Okay, Übersichtlichkeit. Das ist jetzt natürlich ein etwas komplexeres Modell. Würden Sie  
617 sagen, dass Sie noch immer irgendwie das Gesamtsystem auf einen Blick erfassen könnten?  
618 #00:01:10-3#

619 93. B: Ja. Ich würde schon kritischer eigentlich/ Ja, ich habe mir halt überlegt, ob man vielleicht die  
620 ganzen Abbruchkriterien, ob man das bündeln könnte, dass man dann noch sagt, okay, Abbruch  
621 und dann unter welchen Kriterien das eintreten würde, statt das für jedes einzelne Lane macht.  
622 Ja, also ich glaube Übersichtlichkeit würde ich eher eine Vier geben. Aus dem Grund, dass man  
623 ähnliche Sachen meiner Ansicht nach vielleicht noch ein bisschen bündeln könnte. #00:01:49-0#

624 94. I: Das könnte dann eigentlich gleich auch zum nächsten Kriterium passen. Einfachheit. Könnte  
625 man das Modell noch einfacher darstellen? #00:02:00-3#

626 95. B: Ja, wie Sie schon sagten, habe ich im Grunde schon mit beantwortet. Ja, also aus meiner Sicht  
627 könnte man auf jeden Fall die Abbruchkriterien irgendwie bündeln, dass man sagt, unter all

628 diesen Bedingungen kommt es zum Abbruch und nicht für jedes einzelne Lane. Bei dem anderen,  
629 die anderen Sachen würde ich schon fast getrennt lassen. Ja. #00:02:24-9#

630 96. I: Dann die Logik. Entschuldigung. #00:02:31-3#

631 97. B: Ja, Entschuldigung. Vielleicht noch eine Sache dachte ich mir gerade. Also, wenn es dann noch  
632 komplexer wird, das ist jetzt immer noch ein einfaches Beispiel eigentlich mit dem Heizen,  
633 könnte man sich vielleicht so Richtung Objektorientiert, dass man es bündelt und sagt, hier alles,  
634 was wir Richtung Reaktor messen, ist dann wieder ein Strang. Auch wenn man da verschiedene  
635 Sachen misst und die parallel messen kann, jetzt, die Temperatur, den Füllstand, dann macht  
636 man einen Strang Reaktor, das Objekt, und darunter die einzelnen Sachen, die man misst. Das  
637 könnte es bei noch komplexeren Sachen vielleicht noch einfacher machen. Aber hierfür ist es  
638 jetzt okay, also Einfachheit, ja, würde ich auch eine Vier sagen. #00:03:09-9#

639 98. I: Okay. Dankeschön. Okay, Logik. Wird für Sie klar, was parallel und was sequentiell passiert?  
640 #00:03:23-0#

641 99. B: Das wieder. (lacht) #00:03:25-2#

642 100. I: Genau, ja. #00:03:26-5#

643 101. B: Ja, da stehe ich ja so ein bisschen mit auf Kriegsfuß. Im Grunde können die ganzen Sachen  
644 wieder parallel sein, aber es ist ja mit dem Wait verbunden. Hm. (nachdenklich) Da ist ein Wait.  
645 Da ist kein Wait. Also die ganzen Abbruchkriterien, so wie ich das jetzt verstehe, können einfach  
646 zu jederzeit/ könnte man einfach ein Abbruchkriterium auswählen. Das ist so gewünscht, dass  
647 macht ja auch Sinn. #00:04:01-5#

648 102. I: Genau. //B: Oder? Ja. // Das würde auf jeden Fall jedes Mal im Zyklus überprüft werden, ja.  
649 #00:04:05-7#

650 103. B: Okay. Und den Regler (...) Die ganzen Messungen sind jetzt auch parallel, aber sie werden/  
651 nach einer bestimmten Zeit wird abgebrochen, wenn sie nicht durchgelaufen sind. Also dann  
652 wird gar nichts geregelt, oder? #00:04:32-1#

653 104. I: Genau, also die Zykluszeit gibt quasi vor, wenn schon das Messen zu lange dauern würde,  
654 würde das auch abgebrochen werden. Aber aufgrund der Prozesslogik, was hier durch sequential  
655 angegeben ist, müsste auf jeden Fall zuerst die Messung erfolgt sein, bevor Control ausgeführt  
656 werden kann. #00:04:52-8#

657 105. B: Okay. Alle Messungen müssten erfolgt sein? Wenn aber da eins zu lange dauert, dann könnte  
658 nichts bei Control hinein gehen. #00:05:00-6#

659 106. I: Genau, ja. #00:05:01-4#

660 107. B: Okay. Ja, also irgendwie/ Ja. Da habe ich noch zu viele Fragen, wie man merkt. Deshalb würde  
661 ich da eher wieder auf Drei gehen. Durch die Darstellung wird mir das nicht klar genug, würde  
662 ich sagen. Also da brauche ich schon noch die Erklärung dazu. Die Unterscheidung mit dem Wait  
663 oder Cancel und/ es müssen alle Bedingungen erfüllt sein oder es reicht wenn jetzt eins, was in  
664 Control hineingeht, es geschafft hat und so was. Also, das ist mir nicht ganz klar. #00:05:40-5#

665 108. I: Okay. Und dann haben wir noch Erweiterbarkeit. Meinen Sie, dass man eventuell noch etwas  
666 hinzufügen könnte, um den Informationsgehalt zu verbessern? #00:05:53-1#

667 109. B: Ja, also ich überlege halt gerade, wie man das/ was mir fehlt bei dem Fluss auch irgendwie  
668 hinzufügen könnte mit/ Vielleicht irgendwie mit einem Gateway, also was aus den ganzen  
669 Messungen heraus kommt und jetzt sagt, 'Okay, die müssen halt alle erfüllt sein.' So etwas wie  
670 Exclusive Gateway, Parallel Gateway. Die gibt es ja alle schon. Dass man so etwas irgendwie

671 einbaut und sagt, okay, ja. Also alles, was in Control hineingeht muss erfüllt sein. Solche Sachen.  
672 Bestimmte Bündelung irgendwie, wieder. #00:06:42-1#

673 110. I: Okay. Nächste Frage. Wären Sie jetzt nach diesen Prozessbeispielen und nach der Einführung  
674 aufgrund dieser Erweiterungen vielleicht auch bereit, diese Modellierungsmethode in Ihrem  
675 Arbeitsalltag einzuführen, wenn Sie kontinuierliche Prozesse entwickeln würden? #00:07:04-1#

676 111. B: Ja, also, wenn ich was messen und regulieren müsste, dann ja, auf jeden Fall. (lacht)  
677 #00:07:10-0#

678 112. I: Okay. #00:07:11-8#

679 113. B: So ist es, ja. Aber aktuell bin ich ja nicht in der Position, dass ich so etwas modellieren muss,  
680 aber ja. Doch, da wäre es praktisch. #00:07:20-4#

681 114. I: Es folgen jetzt dann auch ein paar Fragen, die im Allgemeinen, also jetzt noch zwei spezielle auf  
682 die beiden Modelle bezogen, und dann ein bisschen allgemeiner auf die Erweiterungen, wie man  
683 sie von der Usability her bewerten könnte. Also ich würde Sie bitten, vielleicht ganz kurz zu  
684 sagen, wie gut beschreiben die Erweiterungen Ihrer Meinung nach ein Kontrollsystem für diese  
685 Beispiele. Und auch wieder mit einer Skala von Eins bis Fünf, wobei Fünf der beste Wert wäre.  
686 Einmal für Modell Eins, also mit dem PI-Regler, das etwas einfachere Beispiel aus MathWorks.  
687 Und dann das zweite Beispiel aus der Siemens Lehrbibliothek, aus den Unterlagen. #00:08:12-5#

688 115. B: Ja, ich finde es soweit ein sehr gutes Kontrollsystem. Wenn es da bisher keines gab, auf jeden  
689 Fall eine sehr sinnvolle Sache, so etwas einzuführen. #00:08:24-8#

690 116. I: Okay, danke. Würden Sie sagen, dass für eine detailliertere Prozessbeschreibung noch etwas  
691 fehlt? Das haben Sie ja eigentlich auch schon am Anfang ein bisschen angeschnitten, dass Ihnen  
692 physikalische Bedingungen und dergleichen fehlen um das System zu beschreiben. #00:08:41-5#

693 117. B: Ja, also wenn das jetzt die einzige Prozessbeschreibung wäre, die man als Stakeholder  
694 irgendwie zur Hand hat, dann, ja, würde mir wie gesagt noch ein bisschen das Big Picture fehlen,  
695 das physikalische, wie ist das Ganze irgendwie/ wie hängt das zusammen? Wie ist das aufgebaut  
696 auch, also. (lacht) Ja, eher so klassische Prozessmodell und nicht jetzt von der Datenfluss-,  
697 Messungssicht her. #00:09:04-6#

698 118. I: Und jetzt kommt eine Frage, die ist glaube ich hier ganz gut geplant oder die können Sie  
699 vielleicht beantworten. Und zwar mit Erfahrung in der Regelungstechnik, als  
700 Wirtschaftsingenieurin, was würden Sie empfehlen, um diese Erweiterungen zu ergänzen, um sie  
701 für andere Ingenieure, Wirtschaftsingenieure, Maschinenbauer, Verfahrenstechniker  
702 interessanter, attraktiver zu machen? #00:09:37-2#

703 119. B: Ja, also wohl auch wieder Sachen, die ich schon gesagt habe. Dass es auf den ersten Blick noch  
704 übersichtlicher ist, also, das mit den Bündelungen. Mit den Gateways. Was war noch?  
705 Regelungstechnik. (lacht) Den Kurs hatte ich einmal, aber es ist schon ein bisschen her. Ja, nein,  
706 ich muss/ da fällt mir leider gerade nicht mehr ein. #00:10:14-6#

707 120. I: Okay. // B: Leider. // Und jetzt kommt/ Nein, passt schon. Danke. Ich verstehe natürlich, ich ja,  
708 weiß, dass das teilweise auch lange her sein kann und dass das auch sehr viel Stoff in einer  
709 Vorlesung meistens ist. Also, ist mir bewusst. Zum Schluss eine etwas umfangreichere Frage. Es  
710 gibt wieder eine Bewertung von Eins bis Fünf. Und zwar, wie gesagt, geht es ein bisschen um die  
711 Usability. Ich möchte Sie zum Schluss genauer zu den Modellen fragen. Und zwar. Ich gehe mal  
712 einfach wieder die einzelnen Fragen mit Ihnen durch. Wie einfach ist in den gezeigten Modellen  
713 nachzuvollziehen, dass die einzelnen Abläufe parallel und unabhängig voneinander laufen? Also  
714 für Sie als Modellierer, für Sie als User. Wie einfach ist es zu definieren, wann eine Anpassung,  
715 eine Regulierung am System erfolgen soll? Wie einfach ist es die maximale Dauer einer  
716 Anpassung zu definieren? Wie einfach ist es zu definieren, unter welchen Bedingungen sämtliche

717 repetitiven Aufgaben beendet werden sollen? Also, Ausbruch aus der Closed Loop. Wie einfach  
718 ist es zu definieren, dass danach Aufräumaufgaben einmalig zu erfolgen haben? Wie einfach ist  
719 es komplexe Abläufe im Kontext von kontinuierlichen Prozessen mit diesen Erweiterungen zu  
720 beschreiben? Also um das Verständnis für Regelprozesse ein bisschen zu verbessern, könnte  
721 man sagen. Erste Frage. Wie einfach ist in den gezeigten Modellen nachzuvollziehen, dass die  
722 einzelnen Abläufe parallel und unabhängig voneinander laufen? #00:12:02-1#

723 121. B: Also aktuell sind sie alle parallel dargestellt. Von daher ist das schon sehr einfach  
724 nachzuvollziehen, würde ich sagen. Fünf. #00:12:16-8#

725 122. I: Okay, danke. Das zweite wäre, wie einfach ist es zu definieren, wann eine Anpassung am  
726 System erfolgt. #00:12:28-8#

727 123. B: Das steht ja im Control-Block, also. #00:12:32-2#

728 124. I: Ja. #00:12:33-1#

729 125. B: Ja. Auch sehr einfach. Ich störe mich ein bisschen an dem Wie. Also, ich habe es jetzt noch  
730 nicht selber angepasst. Wie würde man das denn jetzt denn genau anpassen? (lacht)  
731 #00:12:45-1#

732 126. I: Man würde in //B: Entschuldigung. // Nein, passt schon. Man würde in der [REDACTED] / wie gesagt,  
733 am Anfang habe ich Ihnen ja gezeigt, wenn man das Closed Loop Subsystem in den Prozess  
734 einfügen würde, dann hätte man gleich mal am Anfang die drei Lanes, also, Measure, Control  
735 und Abbruch, also Cancel. Und dann müsste man entsprechend definieren, was man da für Tasks  
736 noch nachher einfügen möchte und die Attribute dazu müsste man definieren. Also beim  
737 Measure zum Beispiel so wie auch bei Control Interval duration/ nein, Entschuldigung, da muss  
738 ich jetzt selber nochmal nachschauen, wie wir das genau genannt haben. Nur um keinen  
739 Blödsinn zu erzählen. #00:13:25-8#

740 127. B: Aber ist gut, ja. Ich scrolle auch gerade mal hoch. Okay. #00:13:31-9#

741 128. I: So, das war etwas weiter oben. (...) Genau. Interval frequency in Hertz. Das heißt, wie oft pro  
742 Sekunde, könnte man sagen. #00:13:47-9#

743 129. B: Okay, ja dann // I: Das könnte man definieren. // würde ich sagen, das ist einfach. Einfach,  
744 weil es in einzelnen Tabs hinterlegt ist, ja. #00:13:59-4#

745 130. I: Also, Sie kennen das vielleicht eh, wenn man mit der [REDACTED] arbeitet und dann eine  
746 Prozessinstanz mal oder ein Modell mal erstellt hat, dann kann man natürlich im linken Fenster  
747 den Graphen sehen und im rechten dann noch etwaige Attribute genauer definieren oder  
748 Sachen hinzufügen, wenn man das möchte. Und im Grunde so würde es halt wie für andere  
749 Symbole oder für andere Tasks ähnlich hier funktionieren. #00:14:24-5#

750 131. B: Okay. #00:14:26-6#

751 132. I: Vielleicht habe ich Sie vorhin überhört. Das heißt, Sie haben hier jetzt schon etwas dazu //B:  
752 Sehr gut würde ich sagen. // kommentiert? Okay. Dankeschön. Wie einfach ist es zu definieren,  
753 unter welchen Bedingungen sämtliche repetitiven Aufgaben beendet werden sollen?  
754 #00:14:52-5#

755 133. B: Ja, also es wäre jetzt eigentlich über das Cancel oder nach einer bestimmten Zeit oder durch  
756 Abbruchkriterien, oder? #00:15:10-0#

757 134. I: Genau, es wären die Abbruchbedingungen beziehungsweise das Intervall, ja. #00:15:14-5#

758 135. B: Dann wieder sehr einfach. Also, sehr gut, Fünf. #00:15:19-7#

---



759 136. I: Also, das Intervall würde eigentlich/ bei Measure und Control würde man dann definieren, wie  
760 lang es maximal dauern darf bis in den nächsten Zyklus übergeht. Aber dann würde noch immer  
761 das Regelungssystem weiterlaufen. Also das Closed Loop Subsystem würde weiterlaufen. Die  
762 Abbruchbedingungen hingegen sind wirklich dafür gedacht, dass man sagt, okay, man muss den  
763 Prozess beenden. #00:15:44-8#

764 137. B: Okay. Also hier sind jetzt nur die //I: Abbruchbedingungen gemeint.// Abbruchbedingungen.  
765 Okay. Ja, die sind ja sehr klar eigentlich, mit den, ja, neuen Notationen da. Ja. #00:15:59-0#

766 138. I: Okay. Und nach diesen Abbruchbedingungen, wie einfach ist es zu definieren, dass danach  
767 Aufräumaufgaben einmalig zu erfolgen haben? #00:16:14-5#

768 139. B: Aufräumaufgaben. Ich weiß, einmal hatten wir, dass es danach in einen manuellen Prozess  
769 geht, als Beispiel. Wäre so etwas eine Aufräumaufgabe? Nein. #00:16:23-7#

770 140. I: Ja, also zum Beispiel, dass man noch etwas/ Also ich weiß, dass dieser Prozess abgebrochen  
771 wird und möchte zum Beispiel einfach den Rührreaktor herunterfahren, also ich möchte eine  
772 Shutdown-Routine zum Beispiel starten. #00:16:42-5#

773 141. B: Okay. #00:16:46-3#

774 142. I: Die würde dann ja auch nur einmalig erfolgen. #00:16:49-5#

775 143. B: Aber jedes Mal wenn der Ausnahmefall eintritt, also, wenn ich den Exit nehmen, dann könnte  
776 man es ja einfach sequenziell dahinter schreiben. Dann wäre es wieder sehr einfach zu  
777 definieren. #00:17:07-0#

778 144. I: Genau, ja. #00:17:07-8#

779 145. B: Ja, okay. Ja, dann sehr gut. #00:17:11-0#

780 146. I: Und das letzte wäre, wie einfach ist es komplexe Abläufe im Kontext von kontinuierlichen  
781 Prozessen, also hier spielt man speziell auf Regelung statt Steuerung an, mit diesen  
782 Erweiterungen zu beschreiben? #00:17:30-0#

783 147. B: Regelung statt Steuerung. Könnten Sie das nochmal kurz erklären? (lacht) #00:17:37-7#

784 148. I: Also, ja, wie gesagt, dass man eher keine diskreten Befehle immer wieder nacheinander  
785 absetzt, sondern dass man hier eine Regelung durchführt. #00:17:48-8#

786 149. B: Also, entsprechend des Verlaufes anpasst. Okay, ja, entsprechend der Messung. #00:17:56-4#

787 150. I: Genau, ja. #00:17:57-0#

788 151. B: Ja, also auf jeden Fall eine Bereicherung. Ich sehe es eher so, mit dem, sobald es sehr komplex  
789 wird, also wenn es dann irgendwann zu viele parallele Stränge nebeneinander sind, wie gesagt,  
790 sehe ich momentan noch Erweiterungsbedarf um bestimmte Bündelung und so etwas, aber,  
791 dementsprechend würde ich da vielleicht auf die Vier mal gehen. Dass es noch gut ist, wegen  
792 dem komplexeren in Klammern. (lacht) #00:18:33-2#

793 152. I: Okay, gut. Dankeschön. Ja, damit sind wir durch. Danke, dass Sie sich Zeit genommen haben.  
794 Und wenn Sie noch Feedback für das Interview haben, also wenn Ihnen etwas aufgefallen ist in  
795 der Fragestellung oder wie die Fragen formuliert sind, dann bin ich hier natürlich auch immer  
796 dankbar für Feedback, dass ich es eventuell auch noch verbessern kann. Ich werde jetzt auch  
797 gleich die Aufnahme stoppen. #00:18:57-4#

---

