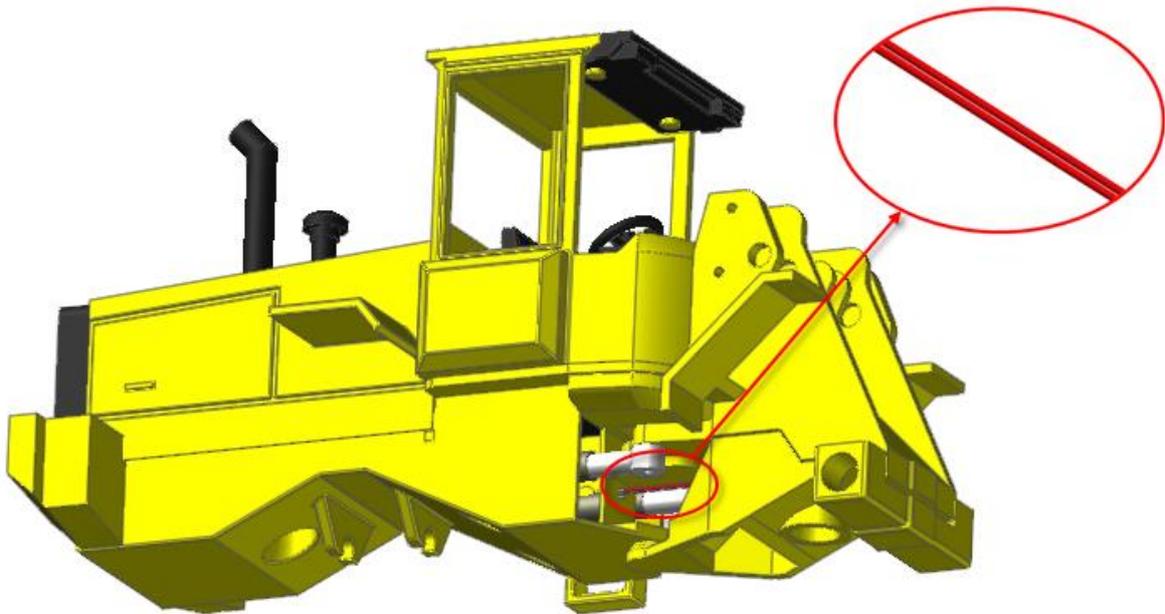




4WD Loader Tutorial (eTemplate)



Copyright © 2020 FunctionBay, Inc. All rights reserved.

User and training documentation from FunctionBay, Inc. is subjected to the copyright laws of the Republic of Korea and other countries and is provided under a license agreement that restricts copying, disclosure, and use of such documentation. FunctionBay, Inc. hereby grants to the licensed user the right to make copies in printed form of this documentation if provided on software media, but only for internal/personal use and in accordance with the license agreement under which the applicable software is licensed. Any copy made shall include the FunctionBay, Inc. copyright notice and any other proprietary notice provided by FunctionBay, Inc. This documentation may not be disclosed, transferred, modified, or reduced to any form, including electronic media, or transmitted or made publicly available by any means without the prior written consent of FunctionBay, Inc. and no authorization is granted to make copies for such purpose.

Information described herein is furnished for general information only, is subjected to change without notice, and should not be construed as a warranty or commitment by FunctionBay, Inc. FunctionBay, Inc. assumes no responsibility or liability for any errors or inaccuracies that may appear in this document.

The software described in this document is provided under written license agreement, contains valuable trade secrets and proprietary information, and is protected by the copyright laws of the Republic of Korea and other countries. UNAUTHORIZED USE OF SOFTWARE OR ITS DOCUMENTATION CAN RESULT IN CIVIL DAMAGES AND CRIMINAL PROSECUTION.

Registered Trademarks of FunctionBay, Inc. or Subsidiary

RecurDyn is a registered trademark of FunctionBay, Inc.

RecurDyn/Professional, RecurDyn/ProcessNet, RecurDyn/Acoustics, RecurDyn/AutoDesign, RecurDyn/Bearing, RecurDyn/Belt, RecurDyn/Chain, RecurDyn/CoLink, RecurDyn/Control, RecurDyn/Crank, RecurDyn/Durability, RecurDyn/EHD, RecurDyn/Engine, RecurDyn/eTemplate, RecurDyn/FFlex, RecurDyn/Gear, RecurDyn/DriveTrain, RecurDyn/HAT, RecurDyn/Linear, RecurDyn/Mesher, RecurDyn/MTT2D, RecurDyn/MTT3D, RecurDyn/Particleworks I/F, RecurDyn/Piston, RecurDyn/R2R2D, RecurDyn/RFlex, RecurDyn/RFlexGen, RecurDyn/SPI, RecurDyn/Spring, RecurDyn/TimingChain, RecurDyn/Tire, RecurDyn/Track_HM, RecurDyn/Track_LM, RecurDyn/TSG, RecurDyn/Valve are trademarks of FunctionBay, Inc.

Edition Note

This document describes the release information of **RecurDyn V9R4**.

목차

| | |
|------------------------------|----|
| 주의사항 | 4 |
| 사전 학습내용..... | 4 |
| 개요 | 4 |
| 목적 | 5 |
| 4WD 유압 호스 모델 열기 | 6 |
| 예상 소요 시간 | 6 |
| RecurDyn 시작하기..... | 7 |
| Contact 자동 생성 | 11 |
| 목적 | 11 |
| 예상 소요 시간 | 11 |
| 생성해야 할 Contact 에 대한 이해 | 12 |
| Solid Contact 생성 | 12 |
| Template Sheet 정의하기 | 15 |
| 시뮬레이션의 실행 | 21 |
| 결과보기 | 21 |
| Segment Contact 정의 | 23 |
| 목적 | 23 |
| 예상 소요 시간 | 23 |
| Segment Contact 정의 | 24 |
| 시뮬레이션의 실행 | 30 |
| 결과보기 | 30 |



주의사항

1. 이 튜토리얼에서 Sheet 에 Excel Data 를 입력할 때, 색상으로 Header Type 과 Parameter Definition 을 구분하고 있는데, 이때 색상은 **Data 값에는 아무런 지장을 주지 않습니다.**
2. 만약 튜토리얼 과정 중에 eTemplate 을 실행했을 경우, 다음 **eTemplate** 을 실행하기 위해서는 **New Model** 을 불러온 뒤, eTemplate 을 불러와야 합니다.
3. 튜토리얼은 각 Chapter 마다 Excel Data 를 복사해서 사용할 수 있도록 만들어져 있습니다. 왼쪽 기호 표시가 되어 있는 부분을 Sheet 에 복사해서 사용하면 됩니다.



사전 학습내용

이 튜토리얼을 시작하기에 앞서, RecurDyn Tutorial 중 ProcessNet 4WD Loader Tutorial (VSTA)을 따라 해 본 뒤, 진행하시길 바랍니다.

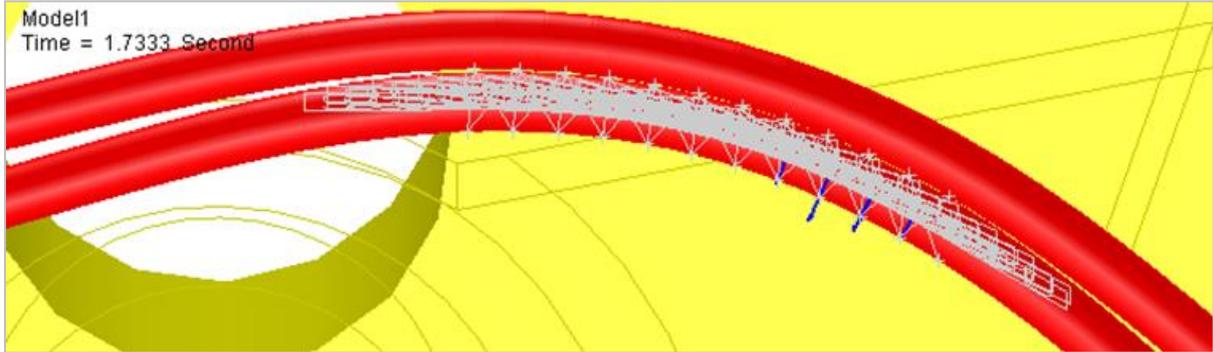
개요

이 튜토리얼에서는 다음과 같은 내용을 배우게 됩니다.

- eTemplate 을 이용하여 일련의 Contact 생성의 자동화
- eTemplate 을 이용하여 더욱 효율적이고 정확한 모델링 작업
- 코드 작업이 아닌, Excel 기반 S4PARK Technology 사용

목적

이 튜토리얼에서는 사륜 구동 로더에 있는 한 쌍의 호스를 시뮬레이션 하게 될 것입니다. 호스는 차량의 앞쪽에 있는 로더 연결 장치의 실린더에 의해 차량의 뒤쪽의 유압 펌프와 연결되며, 로더는 기계관절에 의해 조정됩니다. 호스는 관절이 있는 프레임의 두 부분과 연결되어 구부러지며 서로 Contact 하게 될 것입니다.



어떠한 결함 또는 마모 문제를 예상하기 위해 Contact 을 이해하는 것은 중요합니다. 또한, 실제 호스의 움직임과 Contact 의 위치를 예상하는 것은 어렵기 때문에, 시뮬레이션에서 Contact 이 일어나는 위치와 Contact 힘의 크기를 알기 위해서는 다음 그림과 같이 두 호스의 세그먼트 사이에서 Contact 을 정의해야 합니다. 이 작업은 **다수의 반복작업**으로 인해 지루하게 되고 이로 인해 많은 실수를 초래할 수 있습니다. 그러나, eTemplate 을 이용하면 **더욱 효율적이고 정확한 모델링 작업**을 할 수 있습니다. **ProcessNet** 을 이용하게 되면 **코딩**이라는 복잡한 과정을 거쳐야 하지만



eTemplate 은 간단한 기본 **Data** 를 **Excel** 에 입력하면 됩니다.

Chapter

2

4WD 유압 호스 모델 열기

eTemplate 를 사용하기 위해 모델을 Import 하고 Setting 하겠습니다.



예상 소요 시간

10 분

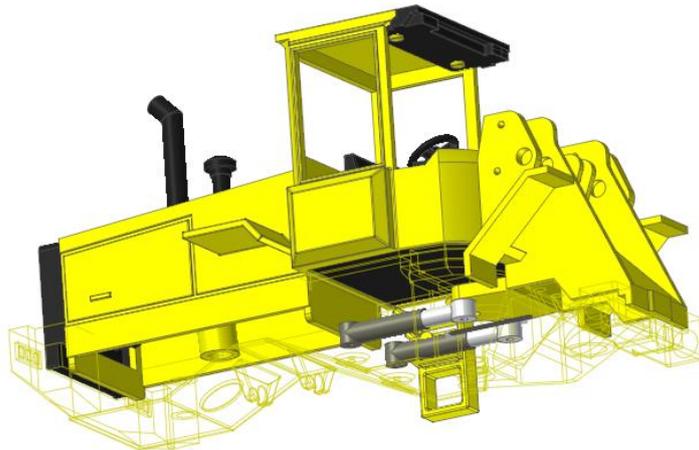
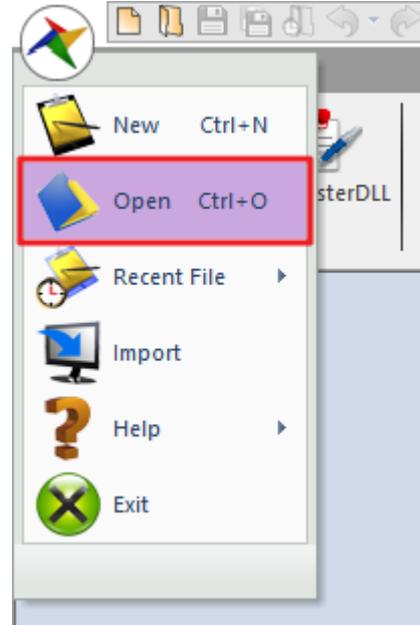
RecurDyn 시작하기

RecurDyn 을 시작한 후 초기 모델 열기



1. 바탕 화면에서, **RecurDyn** 아이콘을 더블 클릭합니다.
2. Start RecurDyn 다이얼로그 박스가 나타나면, 새 모델이 아닌 기존 모델을 사용해야 하므로, 닫아 줍니다.
3. **File** 메뉴에서, **Open** 을 클릭합니다.
4. **eTempalte** 튜터리얼 디렉토리에서
(<Install Dir>
\Help\Tutorial\eTemplate\CreatiomMode\4WDLoader),
4WD_Loader_Start.rdyn 파일을 선택합니다.
5. **Open** 을 클릭합니다.

그리하면, 모델은 다음과 같이 보여 질 것입니다.



빨간 선은 이 튜토리얼에서 다루게 될 유압 호스입니다.

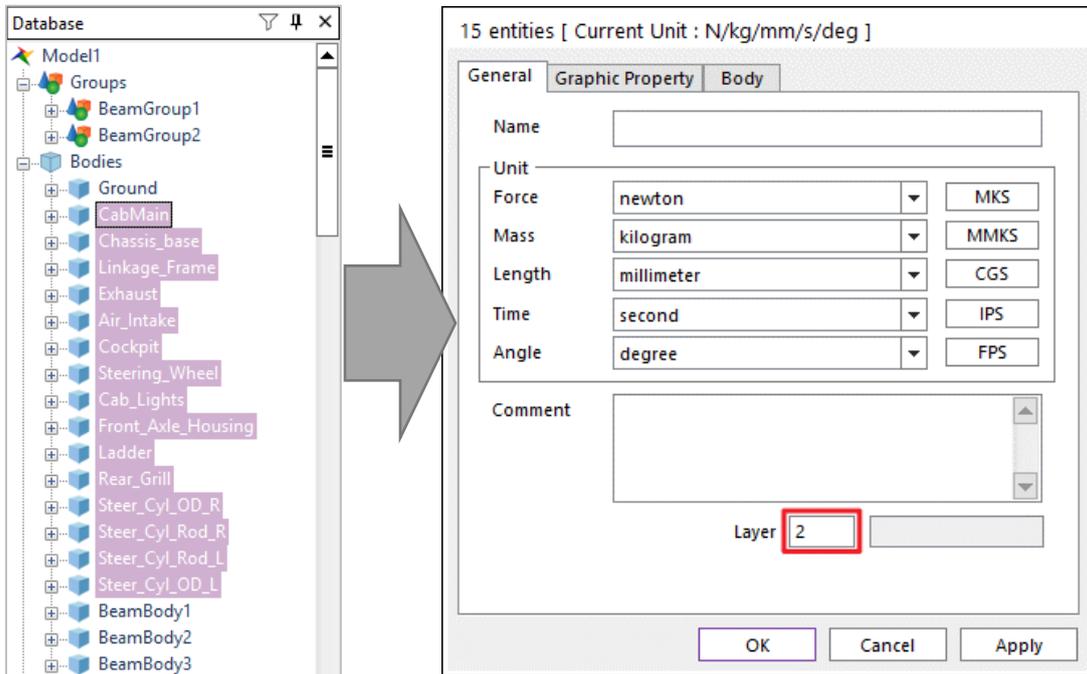
Tip: 호스의 모습이 모호하게 차량의 하단에 보이기 때문에  (Render Each Object) 보기 모드를 선택해야만 위의 그림과 같이 모습이 보여 집니다. (만약 호스의 모습을 보기가 어렵다면, 현재 Shaded 보기 모드인지 확인하여 Render Each Object 보기 모드로 모드를 변경해야 합니다.)

모델 저장하기

1. **File** 메뉴에서, **Save As** 를 클릭합니다.
2. Tutorial 디렉토리에서는 시뮬레이션을 할 수 없기 때문에 다른 디렉토리에 다시 저장합니다.

모델 **Layer Number** 변경

1. 유압 호스를 잘 볼 수 있도록, 유압 호스를 제외한 나머지 바디들의 Layer Number 를 변경합니다.
2. Database 의 Bodies 그룹을 Active 합니다.
3. **CabMain** Body 를 클릭한 뒤, **Shift** Key 를 누른 상태로 **Steer_Cyl_OD_L** Body 를 클릭합니다.
4. 오른쪽 버튼을 클릭하여, Property 를 클릭합니다.
5. **General** Tab 으로 이동하여, Layer Number 를 **2** 로 변경합니다.



6. **OK** 를 클릭합니다.

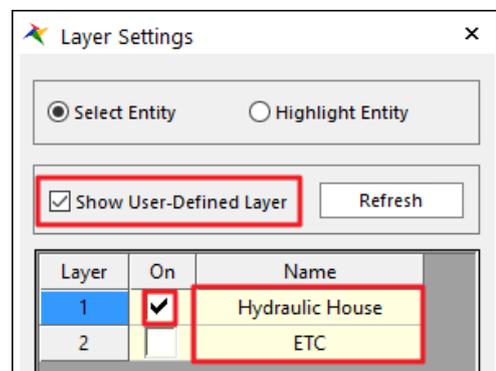


7. **Layer Settings** 아이콘을 클릭합니다.

Layer Settings 대화상자가 열리면 정의된 Layer 만 보이도록 **Show User-Defined Layer** 옵션을 선택합니다.

8. **1** 번 **Layer** 만 보이도록 체크박스를 선택하고 간단하게 이름을 정의합니다.
9. **Layer Settings Dialog** 를 닫습니다.

유압 호스만 Working Window 에 나타납니다.

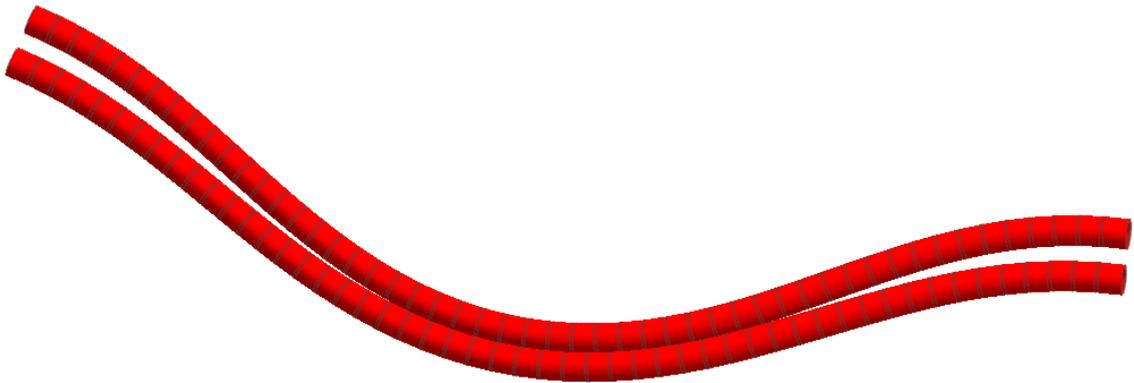




10. **Analysis** 탭의 **Simulation Type** 그룹에서 **Dyn/Kin** 아이콘을 클릭하여 해석합니다.

11. 결과를 확인합니다.

아래의 그림과 같이 변형됩니다. 유압 호스 중간 부분이 서로 Contact 된 것을 볼 수 있습니다. 그래서 3장은 Solid Contact 을 이용하여 유압 호스가 서로 Contact 이 이루어지도록 할



것입니다.

Chapter

3

Contact 자동 생성

목적

유압 호스의 세그먼트 사이에서 Contact 를 생성하는 eTemplate 을 생성하게 될 것입니다.

- RecurDyn 엔티티를 생성하기 위한 정보 수집
- RecurDyn 에 있는 Solid Contact 이용
- Solid Contact Parameter 수정

시뮬레이션을 실행하여 결과를 본 후, 이전보다 향상된 결과를 관찰하기 위해서 Contact 을 다시 생성하게 될 것입니다.

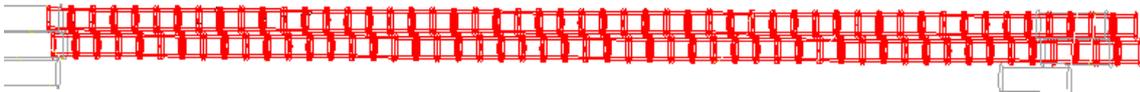


예상 소요 시간

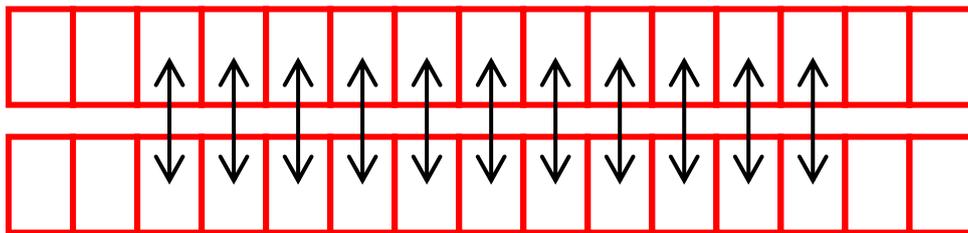
35 분

생성해야 할 **Contact** 에 대한 이해

1 장에서는 아래의 그림과 같이 50 개의 세그먼트를 가지고 있는 두 개의 호스에 대해서 설명하였습니다. 로더와 프레임은 호스 양 끝 지점에 연결되어 있으며, 로더의 회전 중심축이 호스 양 끝 지점 사이에서 적용됩니다. 한 호스에서 관절이 더 잘 굽혀지는 현상이 나타나고 다른 호스는 상대적으로 덜 굽혀지는 현상이 나타나게 됩니다. 두 호스의 굽곡이 다름에 따라 Contact 현상이 호스 중간부분에서 발생하므로 호스의 중간 위치에서 Contact 을 생성합니다.



eTemplate 을 작성하면 아래의 그림처럼 호스의 중간 위치에서 11 개의 contact 을 생성하게 될 것입니다. ProcessNet 을 이용하게 되면 코딩이라는 복잡한 과정을 거쳐야 하지만 eTemplate 은 간단한 기본 Data 를 Excel 에 입력하면 됩니다.

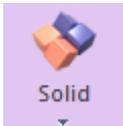
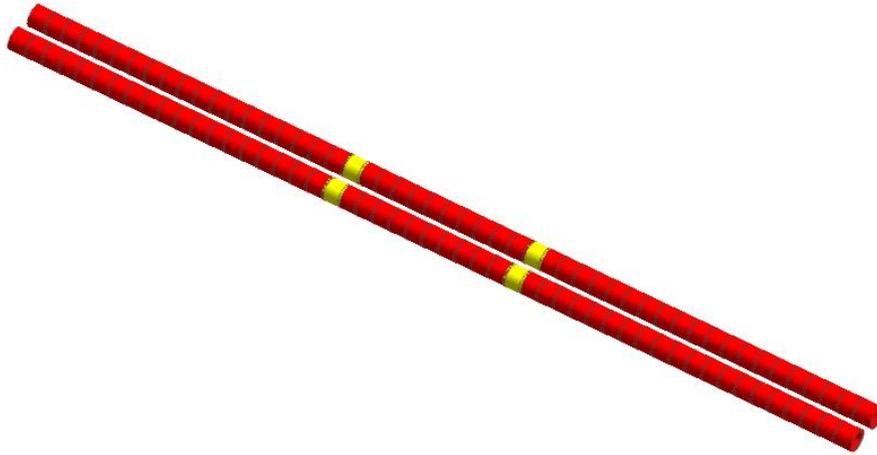


Solid Contact 생성

RecurDyn 엔티티를 생성하기 위한 정보 수집

1. Solid Contact 을 달기 위해서 유압 호스 Body 를 살펴봅니다.
2. 유압 호스는 **BeamGroup1** 과 **BeamGroup2** 로 이루어져 있습니다.
3. BeamGroup1 은 **BeamBody1** 부터 **BeamBody51** 까지 51 개의 Body 를 가지고 있습니다.
4. BeamGroup2 은 **BeamBody52** 부터 **BeamBody102** 까지 51 개의 Body 를 가지고 있습니다.
5. BeamGroup1 와 BeamGroup2 가 Contact 을 이룰 부분을 고려해 봅시다.
6. BeamGroup1 에서는 **BeamBody20** 부터 **BeamBody31** 까지 사용하고, BeamGroup2 에서는 **BeamBody71** 부터 **BeamBody82** 까지 사용하도록 하겠습니다.
7. BeamGroup 에서 각 BeamBody 를 쉽게 알아볼 수 있도록 색상을 변경하도록 하겠습니다.

8. Database 에서 BeamBody 를 마우스 왼쪽 버튼으로 클릭하게 되면 전체가 선택됩니다.
9. Working Window 를 한번 클릭한 뒤 **BeamBody20** 을 마우스 오른쪽으로 클릭하게 되면 BeamBody20 만 선택되고 Property 를 선택할 수 있는 창이 나타납니다.
10. Property 를 선택하고 Graphic Property Tab 으로 이동한 뒤, 색상을 노란색으로 변경합니다.
11. 8~10 번 과정으로 **BeamBody31, BeamBody71, BeamBody82** 에도 적용합니다.



12. **Professional** 탭의 **Contact** 그룹에서 **Solid** 를 선택합니다.
13. **Solid, Solid** 로 모델 생성 옵션이 선택되었는지 확인한 뒤, **BeamBody20** 과 **BeamBody71** 을 클릭합니다.
14. 생성된 Solid Contact Property 를 확인합니다.

| Definition of the Base Solid | |
|---|--|
| Name | BeamBody20.HollowCircularBeam1 Gr |
| Normal Direction | <input checked="" type="radio"/> Up <input type="radio"/> Down |
| <input checked="" type="checkbox"/> Preview Contact Surface | Contact Surface |

| Definition of the Action Solid | |
|---|--|
| Name | BeamBody71.HollowCircularBeam1 Gr |
| Normal Direction | <input checked="" type="radio"/> Up <input type="radio"/> Down |
| <input checked="" type="checkbox"/> Preview Contact Surface | Contact Surface |

15. Base Geometry 로 **BeamBody20.HollowCircularBeam1** 가, Action Geometry 로 **BeamBody71.HollowCircularBeam1** 가 들어가 있음을 확인 할 수 있습니다.

16. eTemplate 작성 시, Contact 정의에서는 Body 이름과 함께 Geometry 도 써줘야 합니다.
17. eTemplate 에서 Contact 을 재생성 할 것이므로 이미 생성된 **SolidContact1** 을 삭제합니다.

Template Sheet 정의하기

Template_Format Sheet 정의하기

Creation Mode 에 사용되는 Template 을 정의합니다.

1. Excel 을 실행하여 **Template_Format** 이라는 이름의 Sheet 를 생성합니다.
Template_Format Sheet 는 템플릿데이터의 처리방법을 정의하는 Sheet 입니다.
2. 생성된 **Template_Format** Sheet 에 Template 형식을 정의하는 **Header** 와 **Parameter** 를 입력합니다.

| Template_Format_Definition | Value |
|----------------------------|----------------------------|
| ModuleKey | S4PARK_Module_Professional |
| TemplateMode | FreeStyleMode |
| UserCommentColumn | 0 |



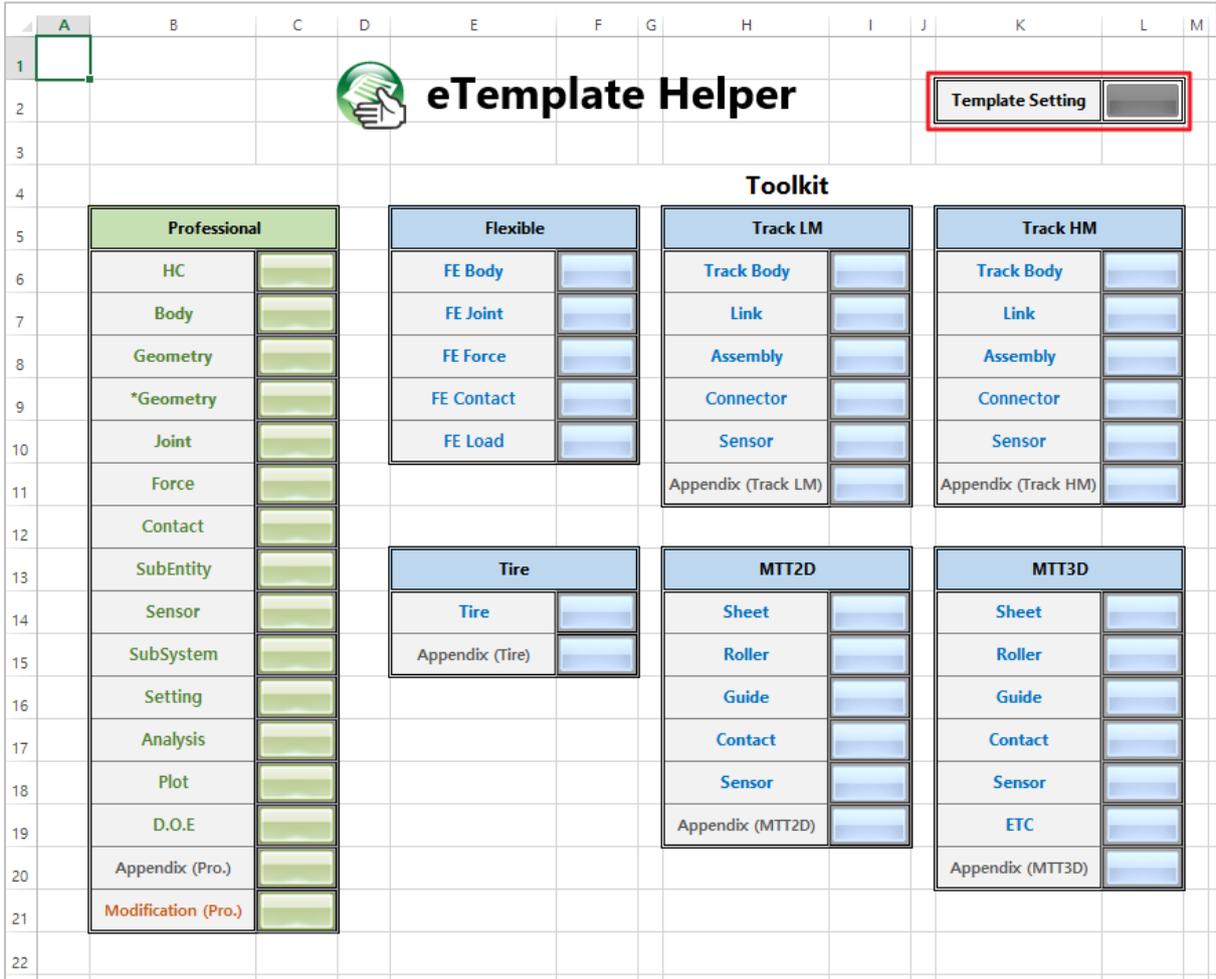
| Template_Format_Definition | Value |
|----------------------------|----------------------------|
| ModuleKey | S4PARK_Module_professional |
| TemplateMode | FreeStyleMode |
| UsercommentColumn | 0 |

- ModuleKey: RecurDyn 제품 모듈 선택
- TemplateMode: Parameter 배열 방법 선택
- UserCommentColumn: 1 에서 5 까지의 값을 입력 받을 수 있으며, Sheet 의 A 열에서 E 열을 사용자 임의로 사용 가능. 사용하지 않을 경우 0 입력

Tip: eTemplate Helper 를 활용한 복사 방법



1. **Customize** 탭의 **eTemplate** 그룹에서 **Helper** 를 클릭하면 **eTemplate Helper** 가



실행됩니다.

2. **Template Setting** 버튼을 클릭하여 Sheet 를 이동합니다.
3. **Template_Format** 의 **Header** 와 **Parameter** 를 복사합니다.
4. 각각의 **Value** 를 교재 내용에 맞게 수정합니다.

앞으로 본 교재를 진행하면서 복사해야 할 Header 와 Parameter 값도 쉽게 복사할 수 있습니다. 또는 완성본으로 제공되는 Template File 을 참조하여 쉽게 복사할 수 있습니다.

(**Template File** 위치:

<Install Dir>
 \Help\Tutorial\eTemplate\CreationMode\4WDLoader \4WD Loader_Template.xlsx)

Template_Data Sheet 정의하기

Creation Mode 를 작동시키는 값들을 입력하기 위해서 **Template_Data** Sheet 를 구성해야 합니다.

1. **Template_Data** 라는 이름의 Sheet 를 생성합니다.
2. 생성된 **Template_Data** Sheet 에 **Solid Contact** 을 생성하는 **Header** 와 **Parameter** 를 입력합니다.

| Header_Contact_Solid | Name | BaseGeometry | ActionGeometry |
|----------------------|------|--------------|----------------|
| Contact_Solid | | | |

Solid Contact 의 각의 **Parameter** 는 아래와 같은 형식의 값들이 입력됩니다.



| Header_Contact_Solid | Name | BaseGeometry | ActionGeometry |
|----------------------|------|--------------|----------------|
| Contact_Solid | | | |

- **Name:** 생성될 Solid Contact 의 이름
 - **BaseGeometry:** 생성될 Solid Contact 의 Base Geometry
 - **ActionGeometry:** 생성될 Solid Contact 의 Action Geometry
3. **Name** 은 **SolidContact1** 부터 **SolidContact11** 까지 하나씩 늘려가면서 넣겠습니다.
 4. Base Geometry 는 앞에서 파악한 정보로부터 BeamGroup1 의 바디인 **BeamBody20.HollowCircularBeam1** 부터 **BeamBody30.HollowCircularBeam1** 를 넣습니다. 이때, 유의할 점은 각 바디가 포함하고 있는 Geometry 인 **HollowCircularBeam1** 을 같이 입력해야 한다는 것입니다.

5. Action Geometry 는 앞에서 파악한 정보로부터 BeamGroup2 의 바디인 **BeamBody71.HollowCircularBeam1** 부터 **BeamBody81.HollowCircularBeam1** 를 넣습니다.



| Header_Contact_Solid | Name | BaseGeometry | ActionGeometry |
|----------------------|----------------|--------------------------------|--------------------------------|
| Contact_Solid | SolidContact1 | BeamBody20.HollowCircularBeam1 | BeamBody71.HollowCircularBeam1 |
| Contact_Solid | SolidContact2 | BeamBody21.HollowCircularBeam1 | BeamBody72.HollowCircularBeam1 |
| Contact_Solid | SolidContact3 | BeamBody22.HollowCircularBeam1 | BeamBody73.HollowCircularBeam1 |
| Contact_Solid | SolidContact4 | BeamBody23.HollowCircularBeam1 | BeamBody74.HollowCircularBeam1 |
| Contact_Solid | SolidContact5 | BeamBody24.HollowCircularBeam1 | BeamBody75.HollowCircularBeam1 |
| Contact_Solid | SolidContact6 | BeamBody25.HollowCircularBeam1 | BeamBody76.HollowCircularBeam1 |
| Contact_Solid | SolidContact7 | BeamBody26.HollowCircularBeam1 | BeamBody77.HollowCircularBeam1 |
| Contact_Solid | SolidContact8 | BeamBody27.HollowCircularBeam1 | BeamBody78.HollowCircularBeam1 |
| Contact_Solid | SolidContact9 | BeamBody28.HollowCircularBeam1 | BeamBody79.HollowCircularBeam1 |
| Contact_Solid | SolidContact10 | BeamBody29.HollowCircularBeam1 | BeamBody80.HollowCircularBeam1 |
| Contact_Solid | SolidContact11 | BeamBody30.HollowCircularBeam1 | BeamBody81.HollowCircularBeam1 |

6. 위의 상태로 Solid Contact 을 생성하게 되면 RecurDyn 기본 값으로 **Parameter** 들이 생성되게 됩니다. 그러나, Solid Contact 에 대한 기본적인 **stiffness** 와 **damping** 값들은 너무 높습니다. 그래서 값을 낮추기 위해 두 **Contact Parameter** 를 설정하는 Option 을 생성합니다.
7. eTemplate 의 Contact 에는 Characteristic 값 중 stiffness 와 damping 을 변경할 수 있는 **ContactProperty** 옵션을 추가 할 수 있습니다.
8. 우선, ContactProperty 에 입력될 값을 정의하기 위해서 **ContactProperty_Info** 이름으로 stiffness 와 damping 을 설정합니다.



| Header_Info_ContactProperty | Name | DampingCoefficient | StiffnessCoefficient |
|-----------------------------|----------------------|--------------------|----------------------|
| Info_ContactProperty | ContactProperty_Info | 0.1 | 1000 |

9. 위에서 정의된 **ContactProperty_Info** 를 Solid Contact 의 **ContactProperty** 옵션으로 입력합니다. 이때 옵션의 위치는 **ActionGeometry** 오른쪽에 생성하여 넣습니다.

10. 시뮬레이션 후, **Force Display** 를 보기 위한 옵션도 함께 넣습니다.



| ContactProperty | ForceDisplay |
|----------------------|--------------|
| ContactProperty_Info | Action |

11. eTemplate 은 아래와 같은 그림처럼 완성됩니다.

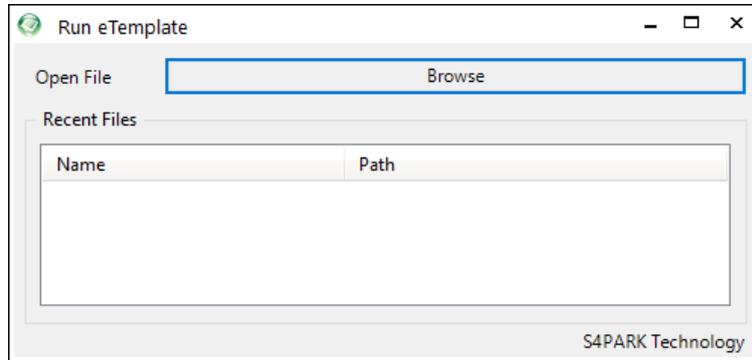
| | A | B | C | D | E | F |
|----|-----------------------------|----------------------|--------------------------------|--------------------------------|----------------------|--------------|
| 1 | Header_Contact_Solid | Name | BaseGeometry | ActionGeometry | ContactProperty | ForceDisplay |
| 2 | Contact_Solid | SolidContact1 | BeamBody20.HollowCircularBeam1 | BeamBody71.HollowCircularBeam1 | ContactProperty_Info | Action |
| 3 | Contact_Solid | SolidContact2 | BeamBody21.HollowCircularBeam1 | BeamBody72.HollowCircularBeam1 | ContactProperty_Info | Action |
| 4 | Contact_Solid | SolidContact3 | BeamBody22.HollowCircularBeam1 | BeamBody73.HollowCircularBeam1 | ContactProperty_Info | Action |
| 5 | Contact_Solid | SolidContact4 | BeamBody23.HollowCircularBeam1 | BeamBody74.HollowCircularBeam1 | ContactProperty_Info | Action |
| 6 | Contact_Solid | SolidContact5 | BeamBody24.HollowCircularBeam1 | BeamBody75.HollowCircularBeam1 | ContactProperty_Info | Action |
| 7 | Contact_Solid | SolidContact6 | BeamBody25.HollowCircularBeam1 | BeamBody76.HollowCircularBeam1 | ContactProperty_Info | Action |
| 8 | Contact_Solid | SolidContact7 | BeamBody26.HollowCircularBeam1 | BeamBody77.HollowCircularBeam1 | ContactProperty_Info | Action |
| 9 | Contact_Solid | SolidContact8 | BeamBody27.HollowCircularBeam1 | BeamBody78.HollowCircularBeam1 | ContactProperty_Info | Action |
| 10 | Contact_Solid | SolidContact9 | BeamBody28.HollowCircularBeam1 | BeamBody79.HollowCircularBeam1 | ContactProperty_Info | Action |
| 11 | Contact_Solid | SolidContact10 | BeamBody29.HollowCircularBeam1 | BeamBody80.HollowCircularBeam1 | ContactProperty_Info | Action |
| 12 | Contact_Solid | SolidContact11 | BeamBody30.HollowCircularBeam1 | BeamBody81.HollowCircularBeam1 | ContactProperty_Info | Action |
| 13 | | | | | | |
| 14 | Header_Info_ContactProperty | Name | DampingCoefficient | StiffnessCoefficient | | |
| 15 | Info_ContactProperty | ContactProperty_Info | 0.1 | 1000 | | |

12. 생성한 **eTemplate** 을 저장합니다.

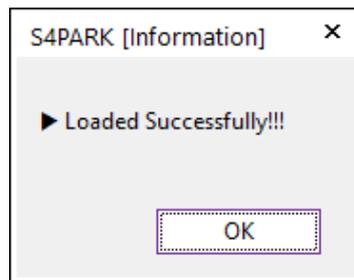
eTemplate 실행



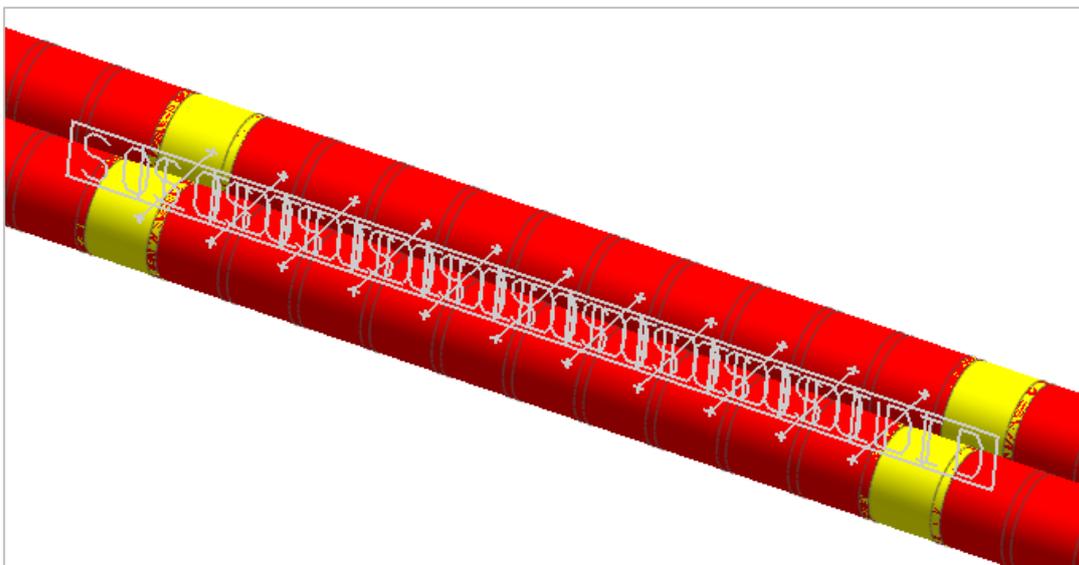
1. **Customize** 탭의 **eTemplate** 그룹에서 **Run** 을 클릭합니다.



2. **Browse** 를 클릭하여, 생성한 eTemplate 파일을 **Import** 합니다.
3. 모델 생성이 완료되었다는 메시지가 나옵니다.



생성된 **Contact** 모델은 아래와 같습니다.

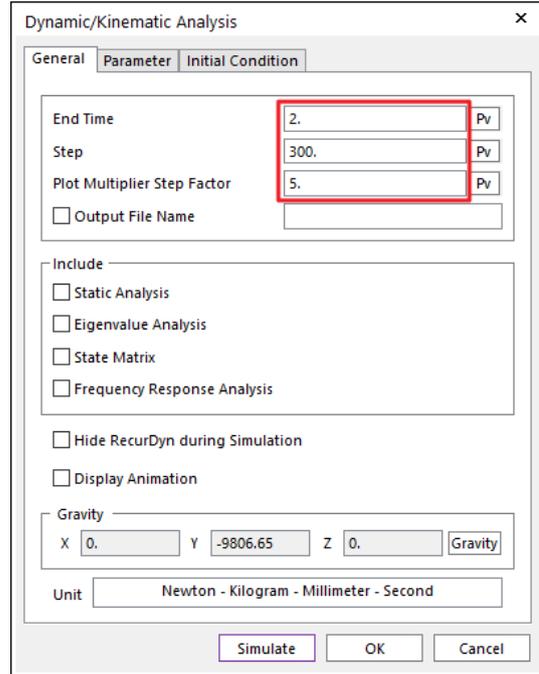


시뮬레이션의 실행

시뮬레이션 실행하기



1. **Analysis** 탭의 **Simulation Type** 그룹에서 **Dyn/Kin** 을 클릭합니다.
2. 시뮬레이션을 실행하기 위해 **End Time** 을 **2.0**, **Step** 을 **300** 으로 설정하고, **Plot Multiplier Step Factor** 는 **5** 로 설정합니다.
3. **Simulate** 를 클릭합니다.

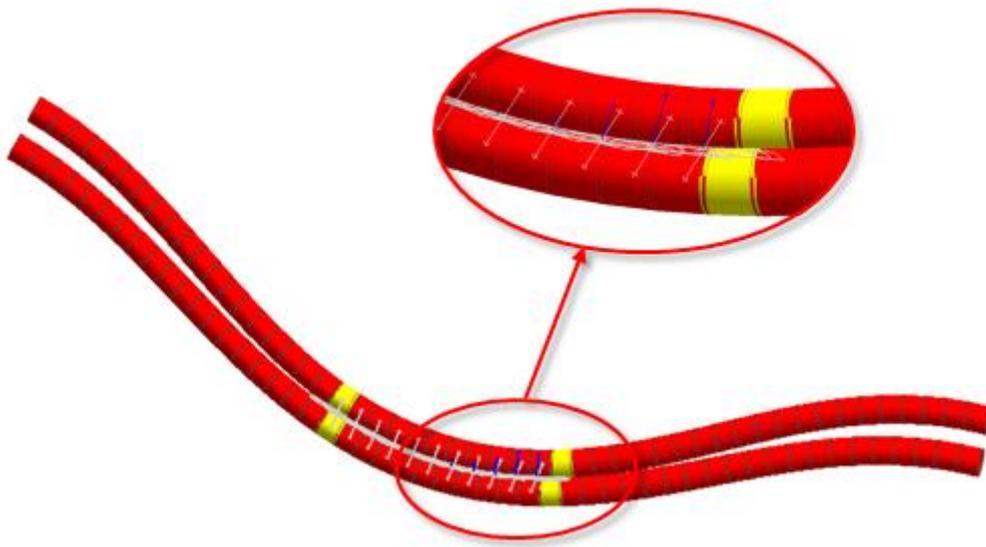


결과보기

결과보기

1. **Analysis** 탭의 **Animation Control** 그룹에서 **Play** 를 클릭합니다.
2. 시뮬레이션의 끝 부분에서 호스는 아래 그림과 같이 변형됩니다.
3. 시뮬레이션의 끝 부분에서 **Force** 가 약간 불규칙적이라는 것을 알 수 있을 것입니다. 움직임을 관찰해보면, 호스가 서로 미끄러지고 있음을 알 수 있습니다.
4. 이것은 해당 세그먼트 사이에서 **Contact** 이 제대로 되지 않아서이며, 이를 위해서 이웃하는 **Segment** 에 **Contact** 을 설정하는 작업이 필요합니다.

다음 장에서는 이웃하는 Segment 에 Contact 을 설정해 보도록 하겠습니다.



Chapter

4

Segment Contact 정의

목적

세그먼트 사이에서 Contact 이 제대로 되지 않아서이며, 이를 위해서 이웃하는 Segment 에 Contact 을 설정하는 작업이 필요합니다.

- eTemplate Data 수정
- Solid Contact 이용하여 Segment 에 Contact 을 설정

시뮬레이션을 실행하여 결과를 본 후, 이전 보다 향상된 결과를 확인해 볼 것입니다.

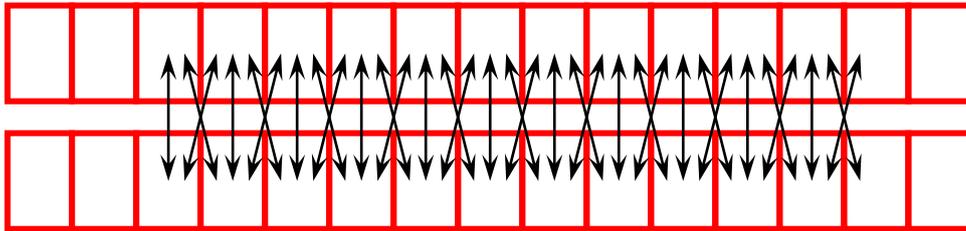


예상 소요 시간

15 분

Segment Contact 정의

세그먼트 사이에서 Contact 이 제대로 되지 않아서 이며, 이를 위해서 이웃하는 Segment 에 Contact 을 설정하는 작업이 필요합니다. 이것은 아래의 그림처럼 보여집니다.



eTemplate Solid Contact 삭제

1. 추가적인 Solid Contact 을 만들기 전에, 3 장에서 생성하였던 Contact 을 삭제합니다.
2. eTemplate 을 Import 할 때, 같은 이름을 가진 Entity 가 있으면 생성을 할 수 없습니다.
3. Database 창에서 첫 번째 Solid contact 인 **SolidContact1** 을 선택합니다.
4. Shift 키를 누른 채로, Database 창에서 마지막 Solid contact 인 **SolidContact11** 을 선택합니다
5. **Delete** 를 눌러 삭제합니다.



eTemplate Solid Contact 추가 생성

1. 3 장에서 생성했던 eTemplate 을 Open 하여 두 종류의 Contact Group 을 추가로 생성하겠습니다.
2. 첫 번째로 유압 호스 **1** 의 **i+1th segment** 와 유압 호스 **2** 의 **jth segment** 사이에 생성된 Contact 에 "a"를 붙여서 이름을 지정합니다.
3. **Base Geometry** 는 앞에서 파악한 정보로부터 BeamGroup1 의 바디인 **BeamBody21.HollowCircularBeam1** 부터 **BeamBody31.HollowCircularBeam1** 를 넣습니다.
4. **Action Geometry** 는 앞에서 파악한 정보로부터 BeamGroup2 의 바디인 **BeamBody71.HollowCircularBeam1** 부터 **BeamBody81.HollowCircularBeam1** 를 넣습니다.

5. Solid Contact 의 eTemplate 정의를 이용하여 아래와 같이 입력합니다.



| Header_Contact_Solid | Name | BaseGeometry | ActionGeometry | ContactProperty | ForceDisplay |
|----------------------|-----------------|--------------------------------|--------------------------------|----------------------|--------------|
| Contact_Solid | SolidContact1a | BeamBody21.HollowCircularBeam1 | BeamBody71.HollowCircularBeam1 | ContactProperty_Info | Action |
| Contact_Solid | SolidContact2a | BeamBody22.HollowCircularBeam1 | BeamBody72.HollowCircularBeam1 | ContactProperty_Info | Action |
| Contact_Solid | SolidContact3a | BeamBody23.HollowCircularBeam1 | BeamBody73.HollowCircularBeam1 | ContactProperty_Info | Action |
| Contact_Solid | SolidContact4a | BeamBody24.HollowCircularBeam1 | BeamBody74.HollowCircularBeam1 | ContactProperty_Info | Action |
| Contact_Solid | SolidContact5a | BeamBody25.HollowCircularBeam1 | BeamBody75.HollowCircularBeam1 | ContactProperty_Info | Action |
| Contact_Solid | SolidContact6a | BeamBody26.HollowCircularBeam1 | BeamBody76.HollowCircularBeam1 | ContactProperty_Info | Action |
| Contact_Solid | SolidContact7a | BeamBody27.HollowCircularBeam1 | BeamBody77.HollowCircularBeam1 | ContactProperty_Info | Action |
| Contact_Solid | SolidContact8a | BeamBody28.HollowCircularBeam1 | BeamBody78.HollowCircularBeam1 | ContactProperty_Info | Action |
| Contact_Solid | SolidContact9a | BeamBody29.HollowCircularBeam1 | BeamBody79.HollowCircularBeam1 | ContactProperty_Info | Action |
| Contact_Solid | SolidContact10a | BeamBody30.HollowCircularBeam1 | BeamBody80.HollowCircularBeam1 | ContactProperty_Info | Action |
| Contact_Solid | SolidContact11a | BeamBody31.HollowCircularBeam1 | BeamBody81.HollowCircularBeam1 | ContactProperty_Info | Action |

6. 두 번째로 유압 호스 1 의 **i th segment** 와 유압 호스 2 의 **j+1 th segment** 사이에 생성된 Contact 에 "b"를 붙여서 이름을 지정합니다.

7. **Base Geometry** 는 앞에서 파악한 정보로부터 BeamGroup1 의 바디인 **BeamBody20.HollowCircularBeam1** 부터 **BeamBody30.HollowCircularBeam1** 를 넣습니다.

8. **Action Geometry** 는 앞에서 파악한 정보로부터 BeamGroup2 의 바디인 **BeamBody72.HollowCircularBeam1** 부터 **BeamBody82.HollowCircularBeam1** 를 넣습니다.

9. Solid Contact 의 eTemplate 정의를 이용하여 아래와 같이 입력합니다.



| Header_Contact_Solid | Name | BaseGeometry | ActionGeometry | ContactProperty | ForceDisplay |
|----------------------|-----------------|--------------------------------|--------------------------------|----------------------|--------------|
| Contact_Solid | SolidContact1b | BeamBody20.HollowCircularBeam1 | BeamBody72.HollowCircularBeam1 | ContactProperty_Info | Action |
| Contact_Solid | SolidContact2b | BeamBody21.HollowCircularBeam1 | BeamBody73.HollowCircularBeam1 | ContactProperty_Info | Action |
| Contact_Solid | SolidContact3b | BeamBody22.HollowCircularBeam1 | BeamBody74.HollowCircularBeam1 | ContactProperty_Info | Action |
| Contact_Solid | SolidContact4b | BeamBody23.HollowCircularBeam1 | BeamBody75.HollowCircularBeam1 | ContactProperty_Info | Action |
| Contact_Solid | SolidContact5b | BeamBody24.HollowCircularBeam1 | BeamBody76.HollowCircularBeam1 | ContactProperty_Info | Action |
| Contact_Solid | SolidContact6b | BeamBody25.HollowCircularBeam1 | BeamBody77.HollowCircularBeam1 | ContactProperty_Info | Action |
| Contact_Solid | SolidContact7b | BeamBody26.HollowCircularBeam1 | BeamBody78.HollowCircularBeam1 | ContactProperty_Info | Action |
| Contact_Solid | SolidContact8b | BeamBody27.HollowCircularBeam1 | BeamBody79.HollowCircularBeam1 | ContactProperty_Info | Action |
| Contact_Solid | SolidContact9b | BeamBody28.HollowCircularBeam1 | BeamBody80.HollowCircularBeam1 | ContactProperty_Info | Action |
| Contact_Solid | SolidContact10b | BeamBody29.HollowCircularBeam1 | BeamBody81.HollowCircularBeam1 | ContactProperty_Info | Action |

| | | | | | |
|---------------|-----------------|--------------------------------|--------------------------------|----------------------|--------|
| Contact_Solid | SolidContact11b | BeamBody30.HollowCircularBeam1 | BeamBody82.HollowCircularBeam1 | ContactProperty_Info | Action |
|---------------|-----------------|--------------------------------|--------------------------------|----------------------|--------|

- 3장에서 정의했던 **ContactProperty_Info** 를 Solid Contact 의 **ContactProperty** 옵션으로 입력합니다. 이때 옵션의 위치는 **ActionGeometry** 오른쪽에 생성하여 넣습니다.
- 시뮬레이션 후, Force 에서 **Force Display** 를 보기 위한 옵션도 함께 넣습니다.
- 입력이 완료되면 다음과 같이 **Template_Data Sheet** 가 완성됩니다.

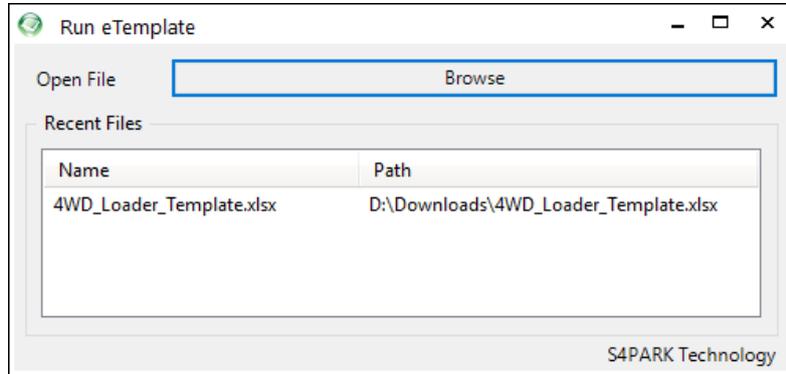
| | A | B | C | D | E | F |
|----|-----------------------------|----------------------|--------------------------------|--------------------------------|----------------------|--------------|
| 1 | Header_Contact_Solid | Name | BaseGeometry | ActionGeometry | ContactProperty | ForceDisplay |
| 2 | Contact_Solid | SolidContact1 | BeamBody20.HollowCircularBeam1 | BeamBody71.HollowCircularBeam1 | ContactProperty_Info | Action |
| 3 | Contact_Solid | SolidContact2 | BeamBody21.HollowCircularBeam1 | BeamBody72.HollowCircularBeam1 | ContactProperty_Info | Action |
| 4 | Contact_Solid | SolidContact3 | BeamBody22.HollowCircularBeam1 | BeamBody73.HollowCircularBeam1 | ContactProperty_Info | Action |
| 5 | Contact_Solid | SolidContact4 | BeamBody23.HollowCircularBeam1 | BeamBody74.HollowCircularBeam1 | ContactProperty_Info | Action |
| 6 | Contact_Solid | SolidContact5 | BeamBody24.HollowCircularBeam1 | BeamBody75.HollowCircularBeam1 | ContactProperty_Info | Action |
| 7 | Contact_Solid | SolidContact6 | BeamBody25.HollowCircularBeam1 | BeamBody76.HollowCircularBeam1 | ContactProperty_Info | Action |
| 8 | Contact_Solid | SolidContact7 | BeamBody26.HollowCircularBeam1 | BeamBody77.HollowCircularBeam1 | ContactProperty_Info | Action |
| 9 | Contact_Solid | SolidContact8 | BeamBody27.HollowCircularBeam1 | BeamBody78.HollowCircularBeam1 | ContactProperty_Info | Action |
| 10 | Contact_Solid | SolidContact9 | BeamBody28.HollowCircularBeam1 | BeamBody79.HollowCircularBeam1 | ContactProperty_Info | Action |
| 11 | Contact_Solid | SolidContact10 | BeamBody29.HollowCircularBeam1 | BeamBody80.HollowCircularBeam1 | ContactProperty_Info | Action |
| 12 | Contact_Solid | SolidContact11 | BeamBody30.HollowCircularBeam1 | BeamBody81.HollowCircularBeam1 | ContactProperty_Info | Action |
| 13 | | | | | | |
| 14 | Header_Info_ContactProperty | Name | DampingCoefficient | StiffnessCoefficient | | |
| 15 | Info_ContactProperty | ContactProperty_Info | 0.1 | 1000 | | |
| 16 | | | | | | |
| 17 | Header_Contact_Solid | Name | BaseGeometry | ActionGeometry | ContactProperty | ForceDisplay |
| 18 | Contact_Solid | SolidContact1a | BeamBody21.HollowCircularBeam1 | BeamBody71.HollowCircularBeam1 | ContactProperty_Info | Action |
| 19 | Contact_Solid | SolidContact2a | BeamBody22.HollowCircularBeam1 | BeamBody72.HollowCircularBeam1 | ContactProperty_Info | Action |
| 20 | Contact_Solid | SolidContact3a | BeamBody23.HollowCircularBeam1 | BeamBody73.HollowCircularBeam1 | ContactProperty_Info | Action |
| 21 | Contact_Solid | SolidContact4a | BeamBody24.HollowCircularBeam1 | BeamBody74.HollowCircularBeam1 | ContactProperty_Info | Action |
| 22 | Contact_Solid | SolidContact5a | BeamBody25.HollowCircularBeam1 | BeamBody75.HollowCircularBeam1 | ContactProperty_Info | Action |
| 23 | Contact_Solid | SolidContact6a | BeamBody26.HollowCircularBeam1 | BeamBody76.HollowCircularBeam1 | ContactProperty_Info | Action |
| 24 | Contact_Solid | SolidContact7a | BeamBody27.HollowCircularBeam1 | BeamBody77.HollowCircularBeam1 | ContactProperty_Info | Action |
| 25 | Contact_Solid | SolidContact8a | BeamBody28.HollowCircularBeam1 | BeamBody78.HollowCircularBeam1 | ContactProperty_Info | Action |
| 26 | Contact_Solid | SolidContact9a | BeamBody29.HollowCircularBeam1 | BeamBody79.HollowCircularBeam1 | ContactProperty_Info | Action |
| 27 | Contact_Solid | SolidContact10a | BeamBody30.HollowCircularBeam1 | BeamBody80.HollowCircularBeam1 | ContactProperty_Info | Action |
| 28 | Contact_Solid | SolidContact11a | BeamBody31.HollowCircularBeam1 | BeamBody81.HollowCircularBeam1 | ContactProperty_Info | Action |
| 29 | | | | | | |
| 30 | Header_Contact_Solid | Name | BaseGeometry | ActionGeometry | ContactProperty | ForceDisplay |
| 31 | Contact_Solid | SolidContact1b | BeamBody20.HollowCircularBeam1 | BeamBody72.HollowCircularBeam1 | ContactProperty_Info | Action |
| 32 | Contact_Solid | SolidContact2b | BeamBody21.HollowCircularBeam1 | BeamBody73.HollowCircularBeam1 | ContactProperty_Info | Action |
| 33 | Contact_Solid | SolidContact3b | BeamBody22.HollowCircularBeam1 | BeamBody74.HollowCircularBeam1 | ContactProperty_Info | Action |
| 34 | Contact_Solid | SolidContact4b | BeamBody23.HollowCircularBeam1 | BeamBody75.HollowCircularBeam1 | ContactProperty_Info | Action |
| 35 | Contact_Solid | SolidContact5b | BeamBody24.HollowCircularBeam1 | BeamBody76.HollowCircularBeam1 | ContactProperty_Info | Action |
| 36 | Contact_Solid | SolidContact6b | BeamBody25.HollowCircularBeam1 | BeamBody77.HollowCircularBeam1 | ContactProperty_Info | Action |
| 37 | Contact_Solid | SolidContact7b | BeamBody26.HollowCircularBeam1 | BeamBody78.HollowCircularBeam1 | ContactProperty_Info | Action |
| 38 | Contact_Solid | SolidContact8b | BeamBody27.HollowCircularBeam1 | BeamBody79.HollowCircularBeam1 | ContactProperty_Info | Action |
| 39 | Contact_Solid | SolidContact9b | BeamBody28.HollowCircularBeam1 | BeamBody80.HollowCircularBeam1 | ContactProperty_Info | Action |
| 40 | Contact_Solid | SolidContact10b | BeamBody29.HollowCircularBeam1 | BeamBody81.HollowCircularBeam1 | ContactProperty_Info | Action |
| 41 | Contact_Solid | SolidContact11b | BeamBody30.HollowCircularBeam1 | BeamBody82.HollowCircularBeam1 | ContactProperty_Info | Action |
| 42 | | | | | | |
| 43 | | | | | | |

- 생성한 **eTemplate** 을 저장합니다.

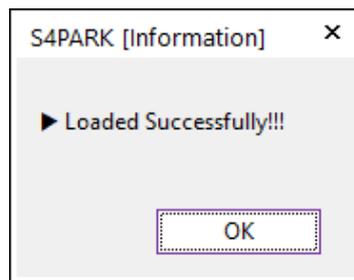
eTemplate 실행



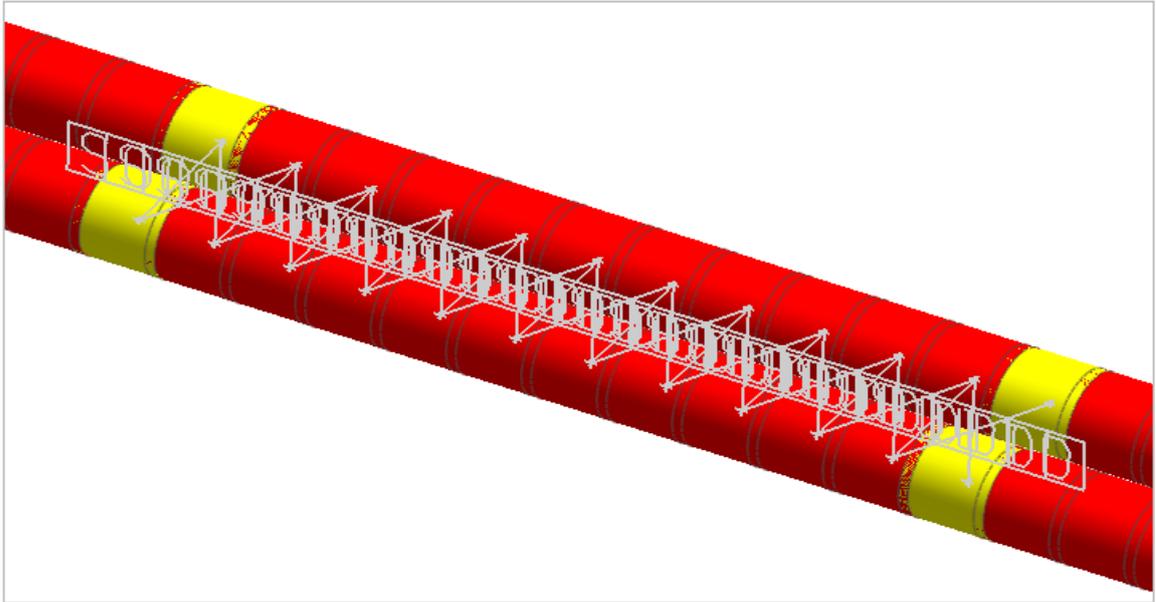
1. **Customize** 탭의 **eTemplate** 그룹에서 **Run** 을 클릭합니다.



2. Recent Files 에 3 장에서 실행했던 eTemplate 기록이 남아있습니다.
3. 이 부분을 더블 클릭하면, 자동으로 eTemplate 파일을 Import 합니다.



4. 모델 생성이 완료되었다는 메시지가 나옵니다.
생성된 Contact 모델은 아래와 같습니다.



시뮬레이션의 실행

시뮬레이션 실행하기

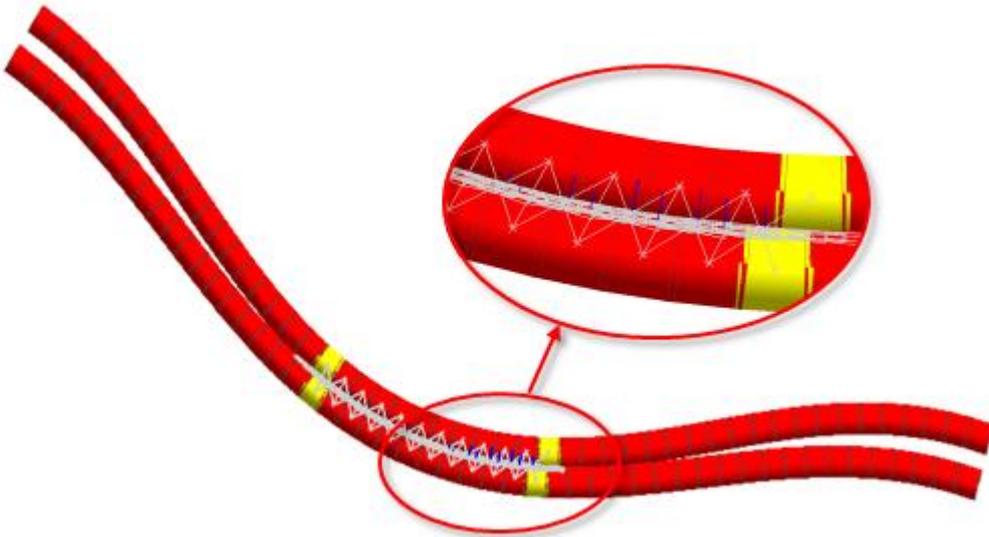


1. **Analysis** 탭의 **Simulation Type** 그룹에서 **Dyn/Kin** 을 클릭합니다.
2. **Simulate** 를 클릭합니다.

결과보기

결과보기

1. **Analysis** 탭의 **Animation Control** 그룹에서 **Play** 를 클릭합니다.
2. 시뮬레이션의 끝 부분에서 호스는 아래 그림과 같이 변형됩니다.
3. Contact Force 가 더욱 매끄러워진 것을 볼 수 있을 것입니다.
4. 지금까지 eTemplate 을 사용하여 복잡한 Contact 을 생성해 보았습니다.



Thanks for participating in this tutorial!