12 APPENDICES

12.1 Appendix A: SMDST User Manual

12.1.1 Version 1.0 - English

12.1.2 Version 1.0 – German

12.2 Appendix B: SMDST Visual Basic Code

12.3 Appendix C: Supplementary CD

12.3.1 SMDST

12.3.2 Traffic-Flow Simulation Model

12.3.3 Input Data: MU_Data.xls

12.3.4 Output Data: Traffic_Results.xls

12.3.5 Station Files: ST01 – ST54

12.4 Appendix D: Simulation Model Application

- Modeling a Scenario -

Supply Medium Decision Support Tool

USER MANUAL

Version 1.0

February 2004

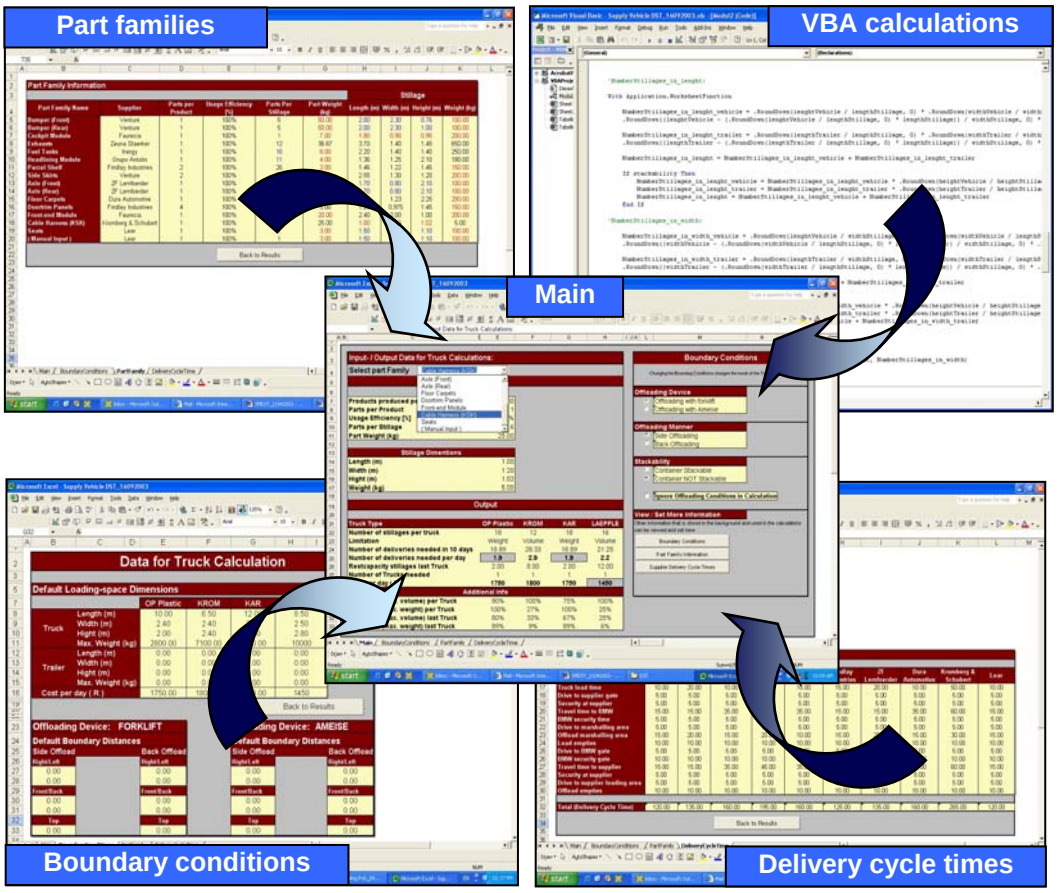


TABLE OF CONTENTS

LIST OF FIGURES

1 INTRODUCTION

This ***decision support tool*** was developed to assist automotive manufacturers in selecting the best supplier transportation medium from various alternatives for the supply of a range of part families. The aim was to create a supportive tool for management, under their own control, that does not attempt to automate the decision process, predefine objectives or impose solutions. It is intended to serve as an extension of the user's problem solving capabilities

2 SYSTEM REQUIREMENTS

The **minimum** system requirements for the effective use of this DST are as follows:

2.1.1 Hardware requirements:

- 10 MB of free storage capacity
- 32 MB RAM
- CD-Rom drive

2.1.2 Software requirements:

- MS-Excel (any version) installed

3 USING THE TOOL

Follow these steps when using the tool for the first time:

Step 1: Open the correct file

1. Open the file "SMDST_English.xls" (found on the installation CD under the following path:

   D:\SMDST\SMDST_v.1.0_English.xls

Step 2: Update / Verify all *input data sheets*:

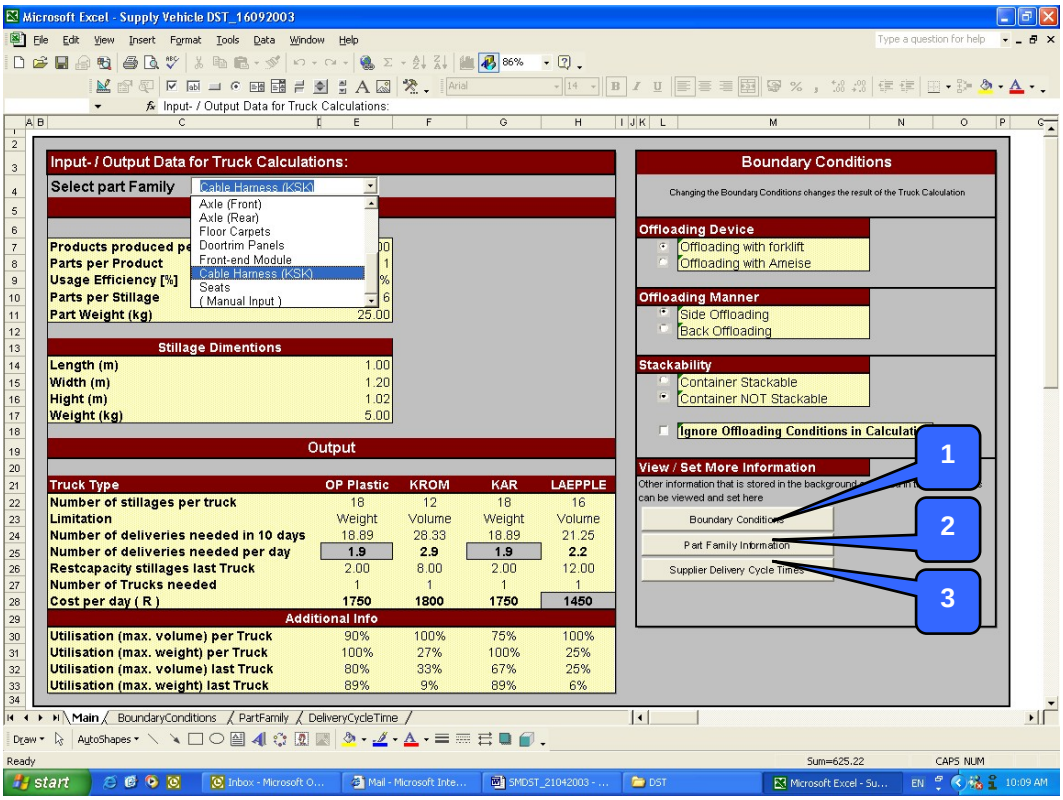


Figure 1: Update input data sheets

1. Click on the **Boundary Conditions** button to go to the boundary conditions input data sheet. Make the required changes and verify that all data displayed on this sheet is correct.

2. Click on the **Part Families** button to go to the part families input data sheet. Make the required changes and verify that all data displayed on this sheet is correct.
3. Click on the **Delivery Cycle Times** button to go to the delivery cycle input data sheet. Make the required changes and verify that all data displayed on this sheet is correct.
4. Click on the **Back to Results** button to return to the Main sheet.

Step 3: Save the changes

1. Save the changes made by selecting "File" (ref. 1 Figure 2), then "Save As" (ref. 2 Figure 2), then specifying the name of the updated file (ref. 3 Figure 3), then selecting "Save" (ref. 4 Figure 3).

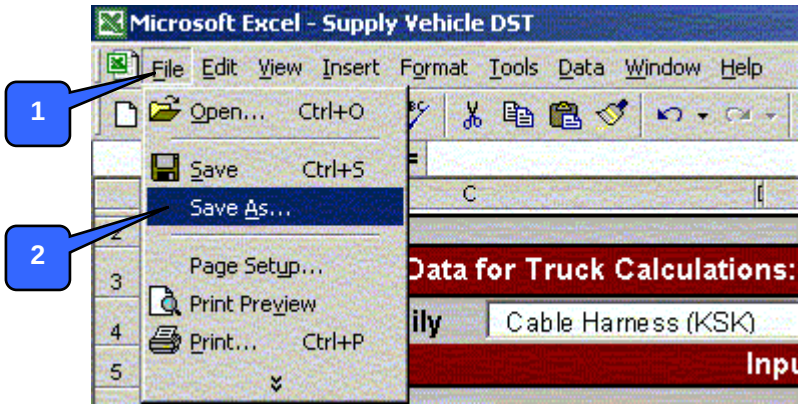


Figure 2: Select "File", then "Save As"

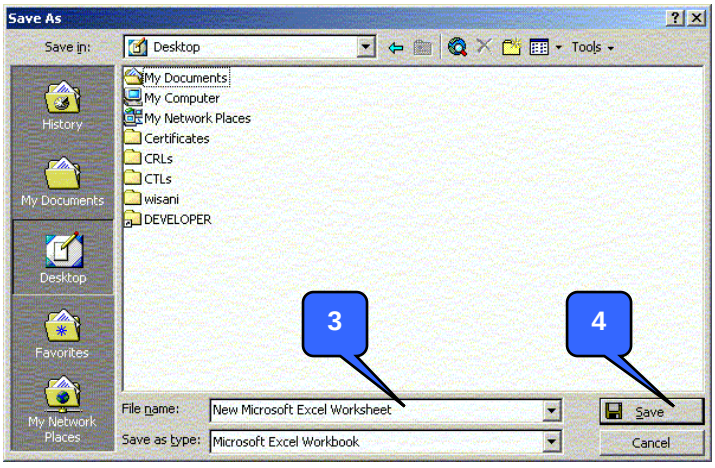


Figure 3: Specify file name and select "Save"

Step 4: Set variables on *main sheet*:

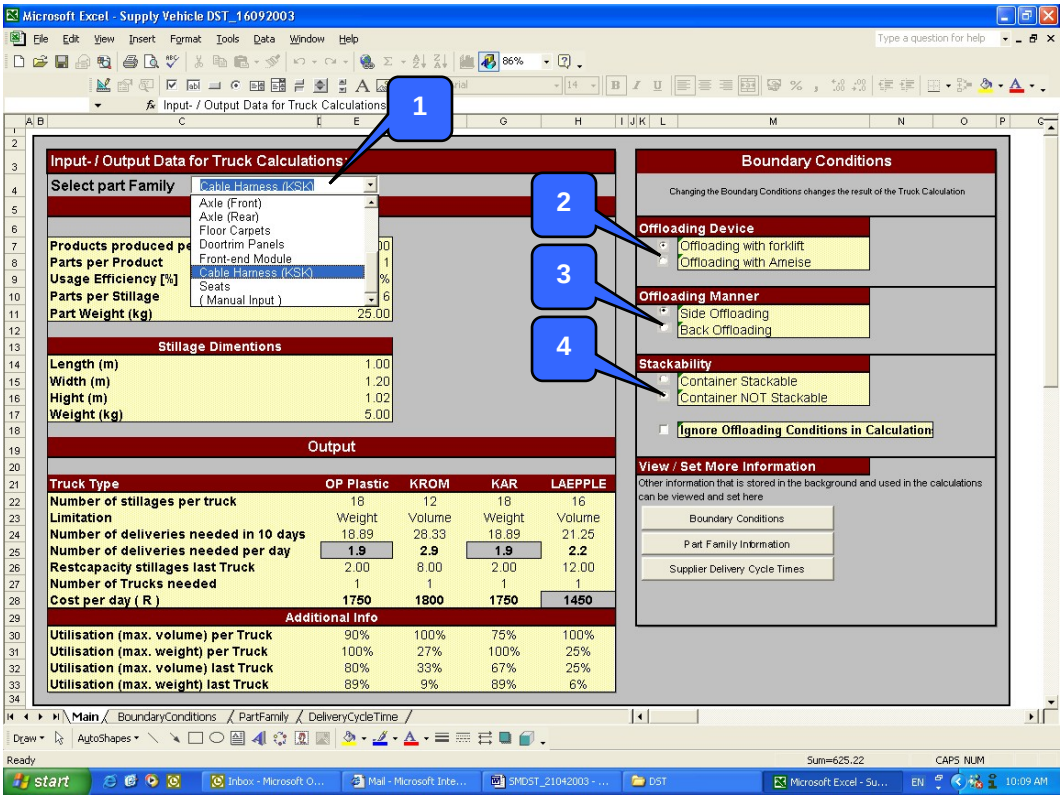


Figure 4: Set variables on main sheet

Set the following criteria on the main sheet:

1. **Part family to be analysed:** select the part family from the drop-down list
2. **Offloading device:** specify whether a forklift or stacker will be used to offload the stillages from the delivery vehicle. (This has an influence on the maximum height that parts can be stacked on the delivery vehicle in later calculations).
3. **Manner of offloading:** specify whether the stillages will be offloaded from the side or back of the delivery vehicle. (This has an influence on the way the parts will be packed on the delivery vehicle).
4. **Stackability:** specify whether the stillages may be stacked on top of one another or not when packed on the delivery vehicle.

Step 5: View and interpret the calculation results:

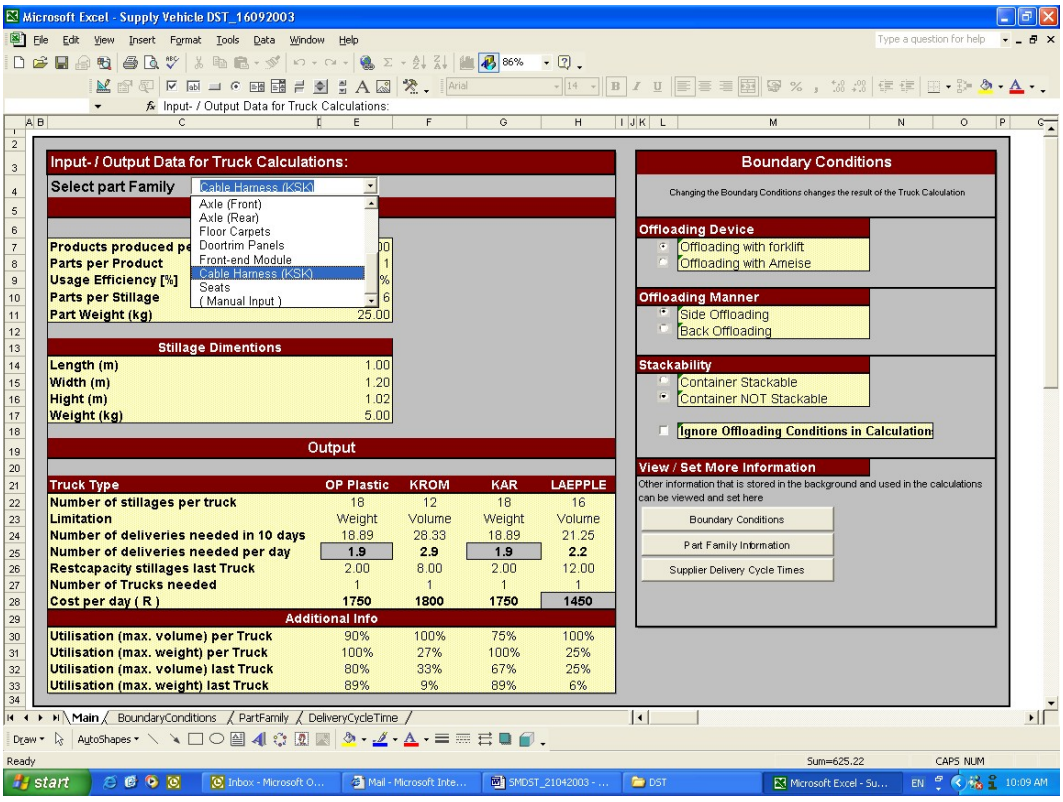


Figure 5: View calculation results

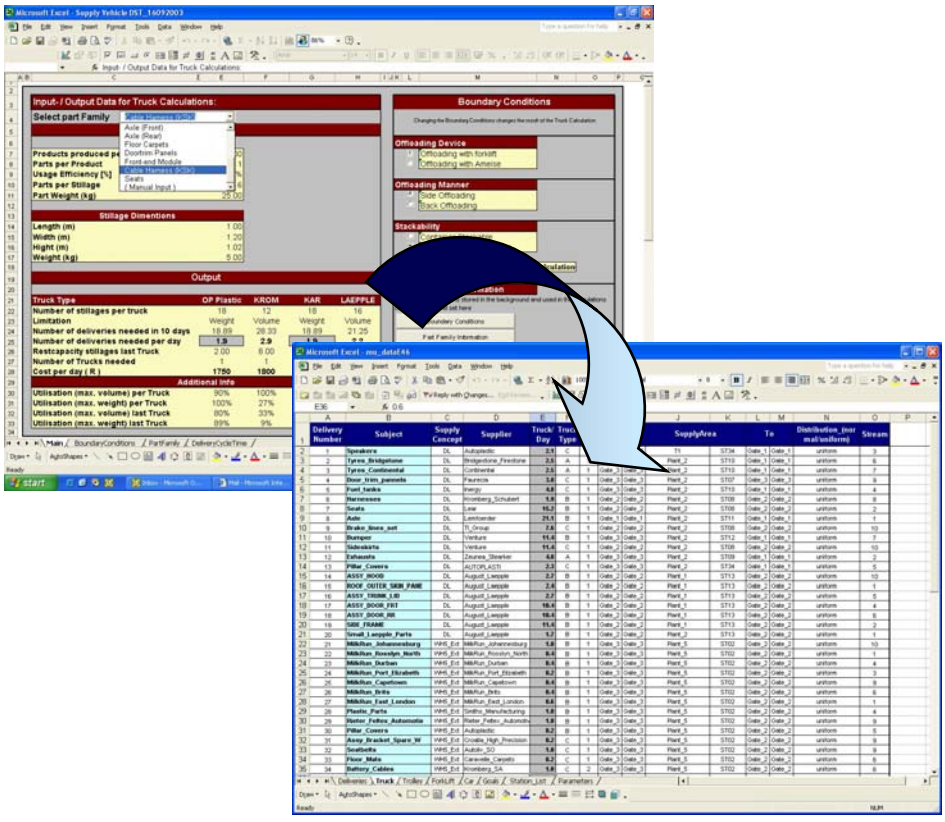
- **Number of stillages per truck:** The calculated maximum number of stillages that can be loaded on a truck
- **Limitation:** The truck loading limitation is displayed as the volume capacity, the weight capacity or both
- **Number of deliveries required per day:** The calculated minimum number of deliveries that will be required with the specific truck per day. The minimum value of the four trucks under consideration for the part family is highlighted for easy recognition
- **Unused capacity stillages of the last truck:** The number of stillages that could have been added to the last delivery to make a "full" delivery

- **Number of trucks required:** The number of trucks required to make the required number of deliveries per day
- **Cost per day:** The cost of using the above mentioned number of trucks per day
- **Additional Information:** Additional information about the utilisation of the specific truck is displayed

Step 6: Decide on delivery vehicle and update traffic flow simulation's input data accordingly:

- With the above mentioned information at hand, the user can decide on the best supply vehicle for the specific part family and update the traffic flow simulation model input data file accordingly.

Supply Medium DST.xls



MU Data.xls

Figure 6: Update traffic flow simulation's input data

4 ENQUIRIES

Please refer any further enquiries related to this tool to:

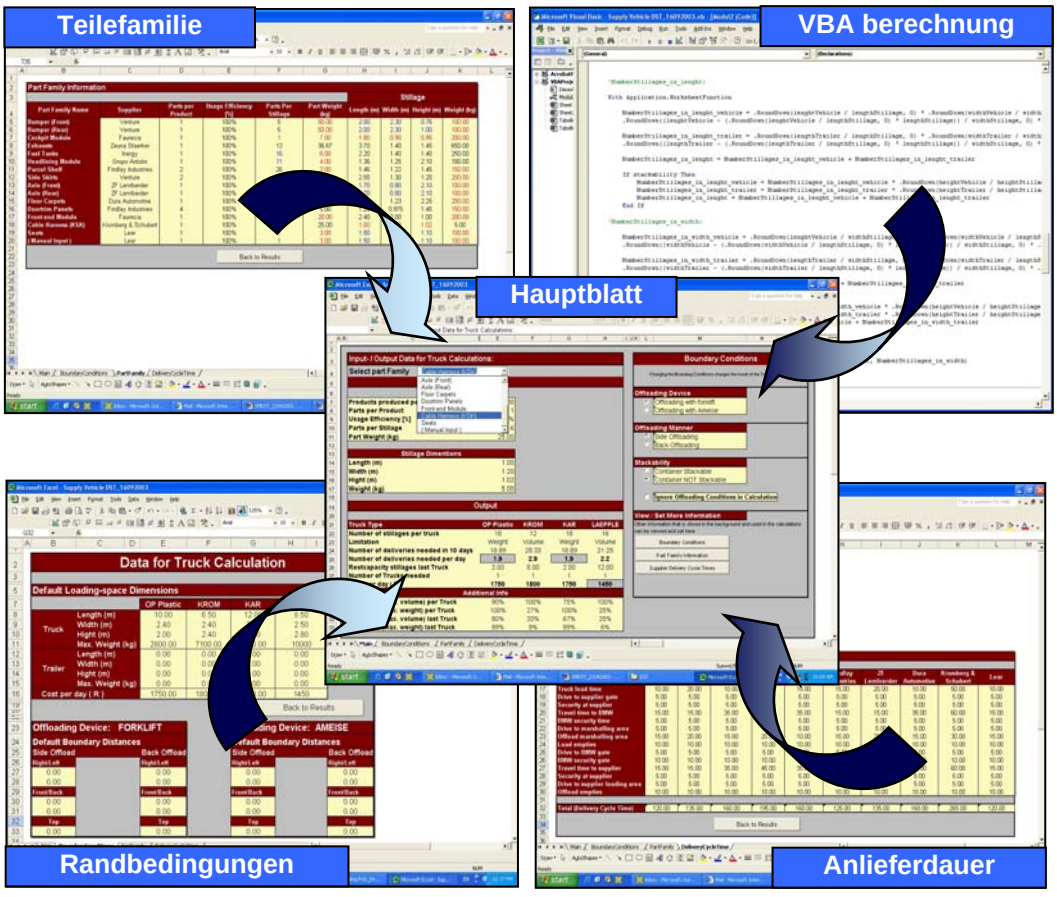
email: piet.vandyk@aidc.co.za

Anlieferungsmethoden Entscheidungsunterstützungstool (SMDST)

Benutzerhandbuch

Version 1.0

Februar 2004



INHALTSVERZEICHNIS

ABBILDUNGSVERZEICHNIS

5 EINLEITUNG

Dieses **Entscheidungsunterstützungstool** wurde zur Unterstützung bei der Auswahl des besten Anliefermediums für eine Reihe von Teilefamilien aus verschiedenen Anlieferalternativen innerhalb der Automobilindustrie entwickelt.
Das Ziel war die Erstellung eines Unterstützungstools für das Management, welches den Entscheidungsprozess nicht versucht zu automatisieren, Objekte vorzudefinieren oder Lösungen aufzuzwingen.
Das Tool soll dem Anwender in dessen Entscheidung nur unterstützend zur Seite stehen.
Es besteht somit vollständige Entscheidungsfreiheit für das Management.

6 SYSTEMANFORDERUNGEN

Die Minimalanforderungen zur effektiven Nutzung diese EUT sind folgende:

1.1.1 Hardwareanforderungen:

- 10 MB Festplattenspeicher
- 32 MB RAM
- CD-Rom Laufwerk

1.1.2 Softwareanforderungen:

- MS-Excel

7 BEDIENUNGSANLEITUNG

Folgende Schritte sind auszuführen, um das Tool nutzen zu können:

Schritt 1: Öffnen der Datei

1. Öffnen sie " SMDST_Deutch.xls " (zu finden auf der Installations-CD unter folgendem Pfad:

   D:\SMDST\SMDST_v.1.0_Deutsch.xls

Schritt 2: Aktualisierung / Prüfung aller Eingabefeder:

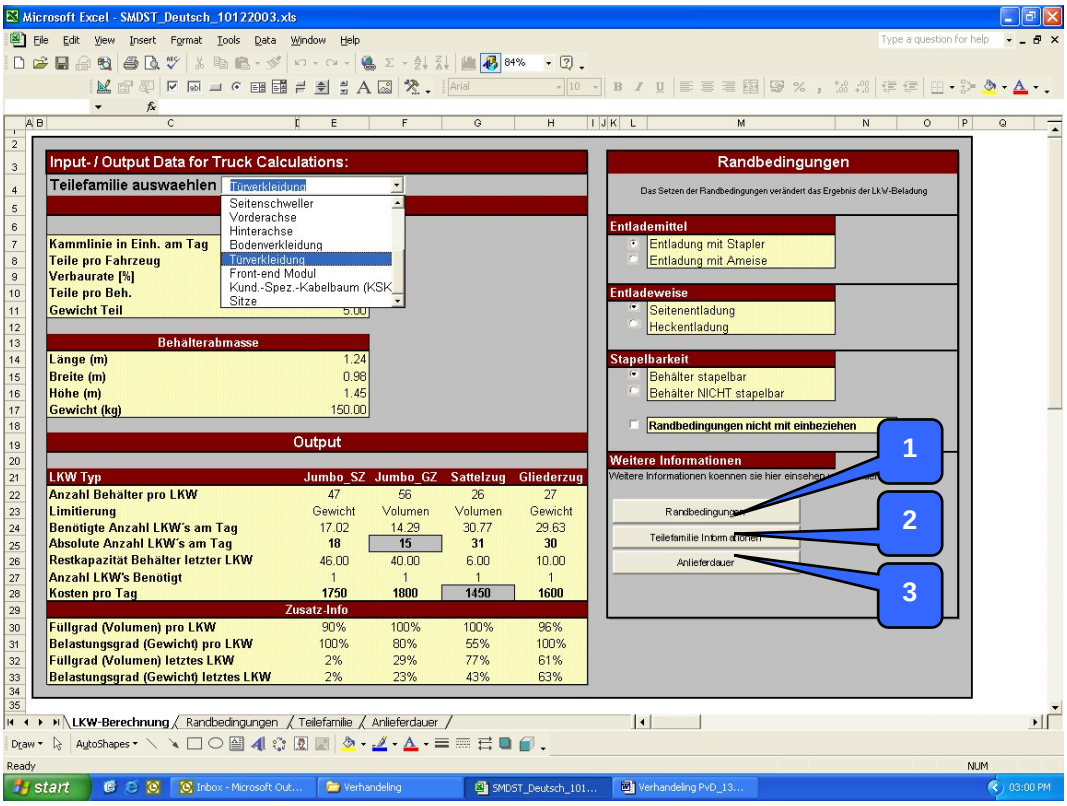


Abbildung 1: Aktualisierung Eingabefelder

1. Klicken sie auf das Feld **Randbedingungen** um zu dem Blatt zu gelangen, wo sie die Grenzbedingungen einzugeben können. Geben sie die erforderlichen Daten ein und versichern sie sich, dass die angezeigten Daten korrekt sind.

2. Klicken sie auf das Feld **Teilefamilie** zur Eingabe der Teilefamiliendaten. Geben sie die erforderlichen Daten ein und versichern sie sich, dass die angezeigten Daten korrekt sind.
3. Klicken sie auf das Feld **Anlieferdauer** zur Eingabe der Behälterumlaufzeit (Lieferumlaufzeit). Geben sie die erforderlichen Daten ein und versichern sie sich, dass die angezeigten Daten korrekt sind.
4. Klinken sie den **Zerück zur Berechnung** Button um zum Hauptblatt zurückzukehren.

Schritt 3: Speicherung der Änderungen

1. Um die Änderungen zu speichern gehen sie im Menüfeld unter "Datei" (siehe Referenz. 1 in Abbildung 2), danach "Speichern unter" (siehe Referenz. 2 in Abbildung 2), danach vergeben Sie einen Namen der aktualisierten Datei (siehe Referenz. 3 in Abbildung 3), danach "Speichern" (siehe Referenz. 2 in Abbildung 3).

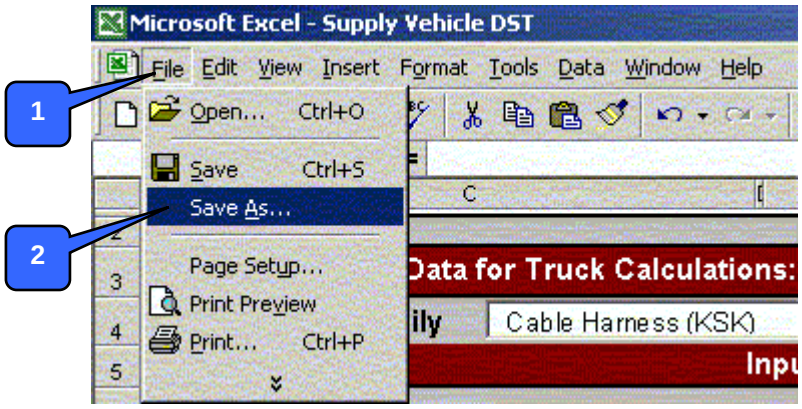


Abbildung 2: Auswahl "Datei", danach "Speichern unter"

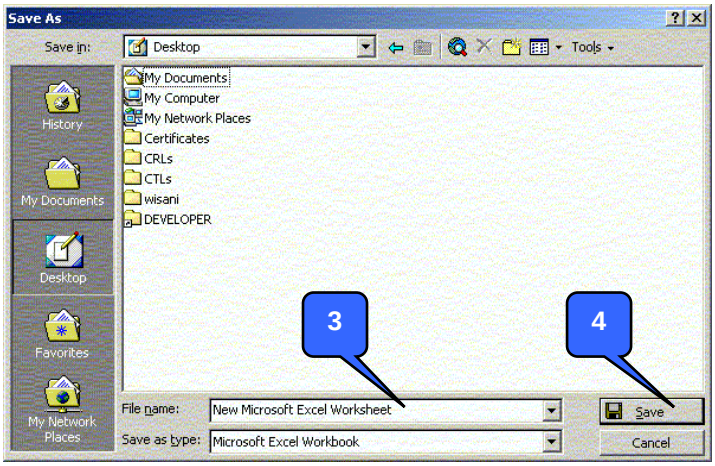


Abbildung 3: Vergeben sie einen Namen und klicken sie "Speichern"

Schritt 4: Setzten der Variablen im "*Hauptblatt*":

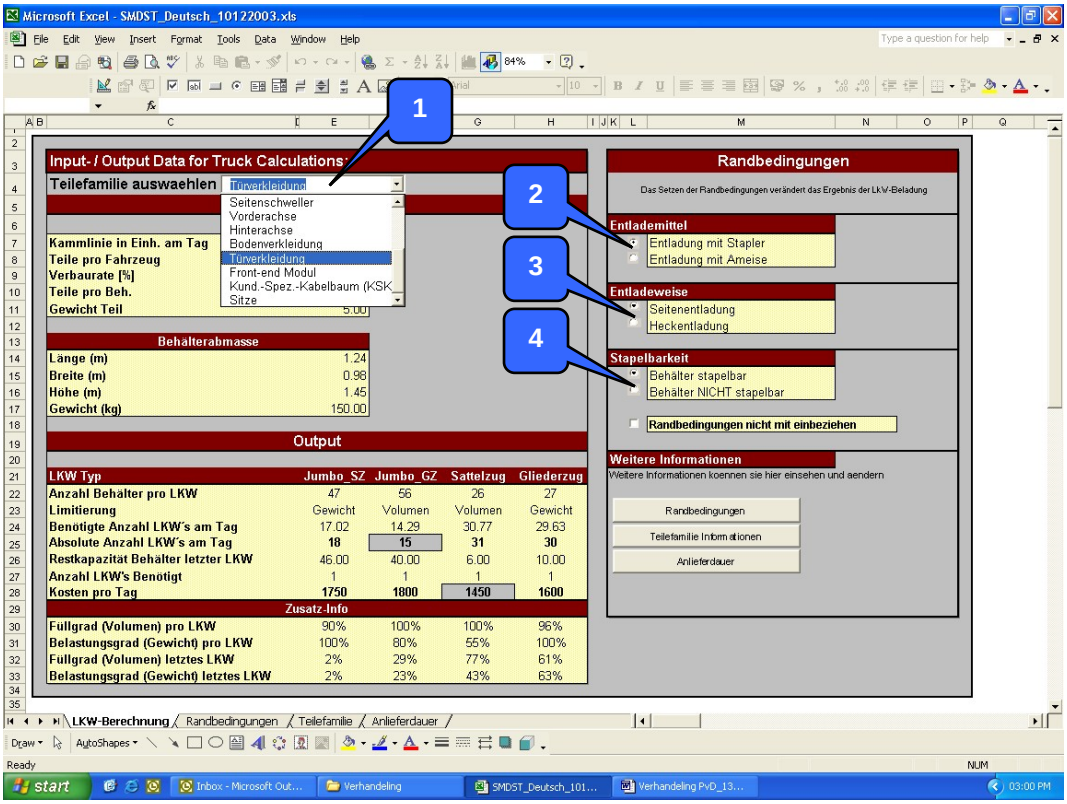


Abbildung 4: Setzen der Variablen im "Hauptfeld"

Setzen sie folgende Kriterien im "Hauptblatt":

1. **Zu analysierende Teilefamilien:** Auswahl der Teilefamilie aus "drop-down" Liste
2. **Abladevorrichtung:** spezifizieren sie ob ein Gabelstapler oder eine Ameise zum abladen der Behälter benutzt wird. (Dies hat Einfluss auf die max. Stapelhöhe der Behälter im LKW für spätere Berechnungen).
3. **Art der Abladung:** spezifizieren sie ob der Behälter von der Site oder von hinten abgeladen wird. (Dies hat Einfluss, wie die Behälter in den LKW geladen werden).
4. **Stapelbarkeit**: spezifizieren sie ob der Behälter stapelbar ist.

Schritt 5:

Ansicht und Interpretation der Berechnungsergebnisse:

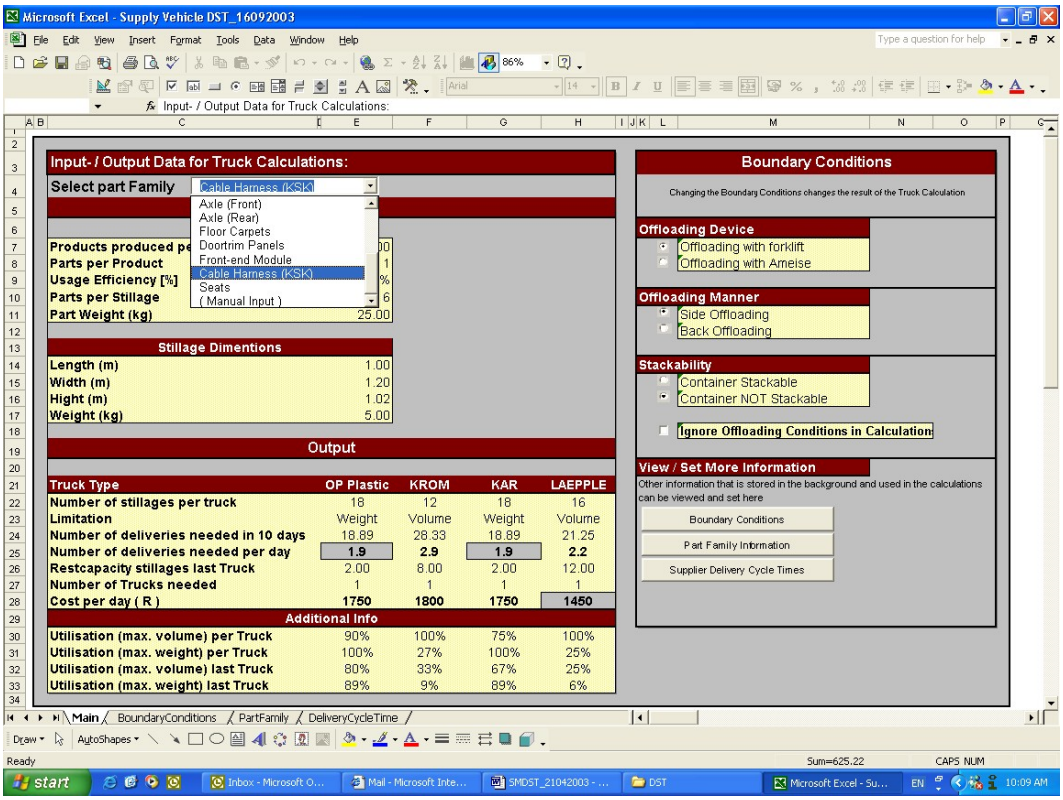


Abbildung 5: Ansicht der Berechnungsergebnisse

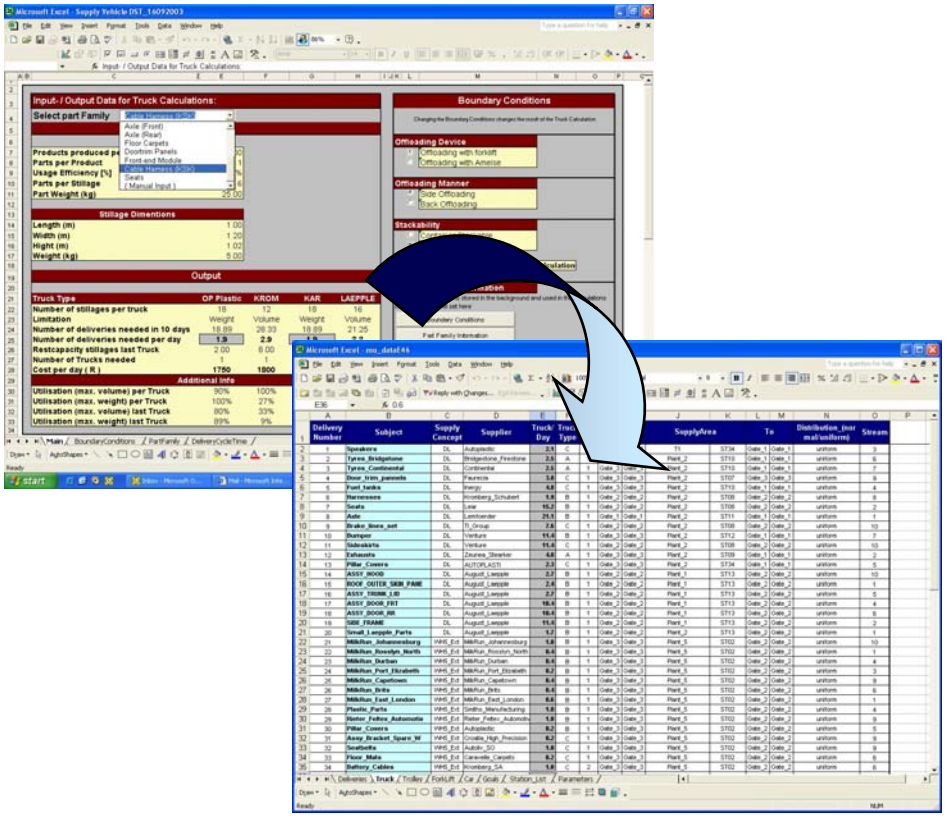
- **Anzahl der Behälter pro LKW:** Die berechnete Maximalanzahl an Behälter können auf den LKW geladen werden.
- **Einschränkung:** Die LKW-Beladungsbeschränkung wird anhand der Volumenkapazität, Gewichtskapazität oder beiden angezeigt.
- **Anzahl der Anlieferungen pro Tag:** Die berechnete Minimalanzahl an Anlieferungen pro Tag die mit dem ausgewählten LKW durchgeführt werden müssen. Innerhalb der 4 angezeigten Möglichkeiten ist das Minimum hervorgehoben.
- **Ungenutzte Ladekapazität im letzten LKW:** Anzahl der Behälter, welchen noch auf den letzten LKW gepasst hätte, wenn er voll beladen werden würde.
- **Anzahl benötigter LKWs:** Anzahl der LKWS die benötigt werden, um die täglich benötigten Anlieferungen abzudecken.

- **Kosten pro Tag:** Kosten für die Verwendung der oben erwähnten Anzahl an LKWs pro Tag.
- **Zusätzliche Informationen:** Zusätzliche Informationen über die Auslastung von einem bestimmten LKW wird angezeigt.

Schritt 6: Entscheidung anhand Anlieferungen und dessen Aktualisierung in die Verkehrssimulations-eingangsdaten:

- Mit den aus der Berechnung erhaltenen Daten muss der Benutzer sich für das beste Anlieferungsfahrzeug für eine bestimmte Teilefamilie entscheiden und die daraus gewonnenen Daten als Eingangsdaten in die Verkehrssimulation einfließen lassen.

SMDST_Deutsch.xls



MU Data.xls

Abbildung 6: Aktualisierung Verkehrssimulationseingangsdaten

8 ANFRAGEN, PROBLEME

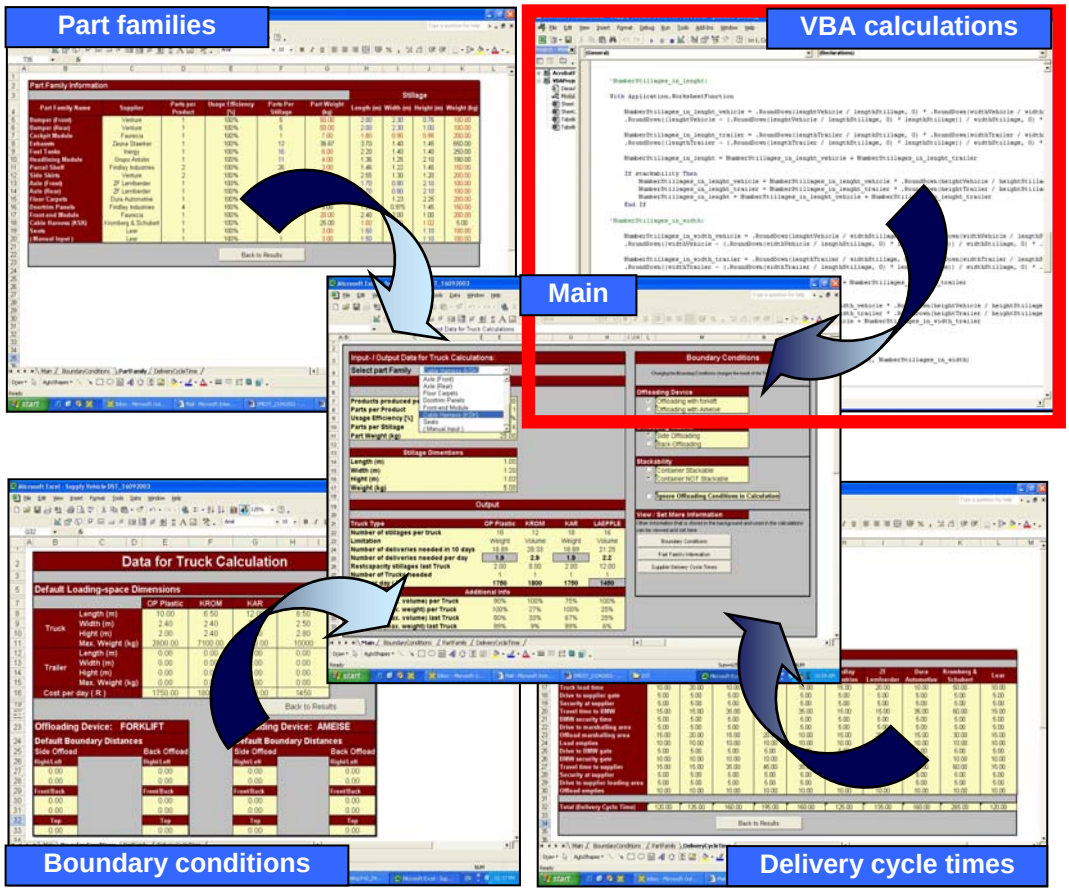
Alle auftretenden Anfragen/Probleme bitte an folgende Adresse senden:

Email: piet.vandyk@aidc.co.za

Appendix B

Supply Medium Decision Support Tool

Visual Basic Code



Following *8.4.1 Calculations in VBA* is the complete VBA code. This can also be viewed by

1. Opening the "SMDST_English.xls" file on the supplementary CD under the path: D:\SMDST\SMDST_v.1.0_English.xls
2. Selecting Tools / Macro / Visual Basic Editor (see Figure I below). The code can then be viewed under “module 2” in the Visual Basic Editor

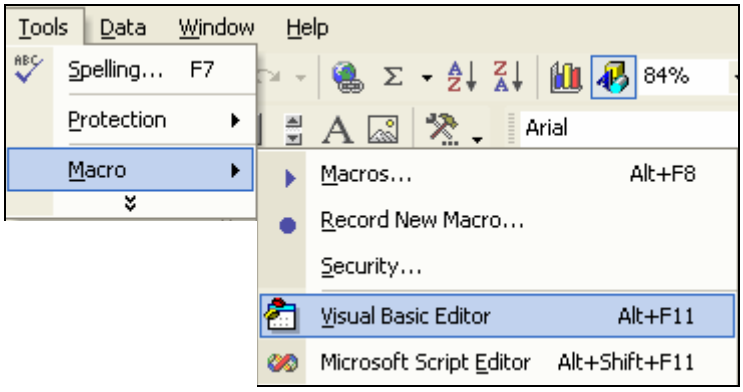


Figure A: Opening Visual basic Editor

```
'Option Explicit
Sub main()

'Declarations:
    Dim Products_produced_per_day As Integer
    Dim Parts_per_Product As Integer
    Dim Efficiency As Double
    Dim Parts_per_Stillage As Integer
    Dim partWeight As Double
    Dim lengthStillage As Double
    Dim widthStillage As Double
    Dim heightStillage As Double
    Dim offloadingDevice As String
    Dim offloadingManner As String
```

```
    Dim stackability As Boolean
    Dim Parts_per_day As Double
    Dim Stillages_per_day As Integer
    Dim NumberStillagesPerVehicle_Volume As Integer
    Dim NumberStillagesPerVehicle_Weight As Integer
    Dim NumberStillagesPerVehicle As Integer
    Dim maxWeightVehicle As Double
    Dim maxWeightTrailer As Double
    Dim stillageWeight As Double
    Dim stillage_with_Parts_Weight As Double
    Dim Deliveries_required_per_day As Double
    Dim Absolute_Deliveries_required_per_day As Integer
    Dim Restcapacity_stillage_last_vehicle As Integer
    Dim temp As Double

'Variables
With Sheets("Main")
   Products_produced_per_day = .Range("E7").Value
   Parts_per_Product = .Range("E8").Value
   Efficiency = .Range("E9").Value
   Parts_per_Stillage = .Range("E10").Value
   partWeight = .Range("E11").Value

   lengthStillage = .Range("E14").Value
   widthStillage = .Range("E15").Value
   heightStillage = .Range("E16").Value
   stillageWeight = .Range("E17").Value

   stillage_with_Parts_Weight = stillageWeight + (partWeight *
   Parts_per_Stillage)

'Test length vs. width
```

```
    If widthStillage > lengthStillage Then
        temp = lengthStillage
        lengthStillage = widthStillage
        widthStillage = temp
    End If

    offloadingDevice = .Range("N9").Value
    offloadingManner = .Range("N13").Value
    stackability = .Range("N15").Value

End With

With Application.WorksheetFunction
    Parts_per_day = Products_produced_per_day *
     Parts_per_Product * Efficiency
     Stillages_per_day = .RoundUp(Parts_per_day /
     Parts_per_Stillage, 0)

     NumberStillagesPerVehicle =
     getNumberStillagesPerVehicle("Jumbo_GZ", lengthStillage,
     widthStillage, heightStillage, offloadingDevice, offloadingManner,
     stackability, 10000, 0, stillage_with_Parts_Weight)

     Deliveries_required_per_day =
     getDeliveries_Required_per_Day(Stillages_per_day,
      NumberStillagesPerVehicle)
      Absolute_Deliveries_required_per_day =
     .RoundUp(Deliveries_required_per_day, 0)
      Restcapacity_stillage_last_vehicle =
     (Absolute_Deliveries_required_per_day –
     Deliveries_required_per_day) * NumberStillagesPerVehicle
```

```
        University of Pretoria etd – Van Dyk, P J S  (2005)
End With
End Sub

Function getNumberStillagesPerVehicle(typeTruck As String,
lengthStillage As Double, widthStillage As Double, heightStillage As
Double, offloadingDevice As String, offloadingManner As String,
stackability As Boolean, maxWeightVehicle As Double,
maxWeightTrailer As Double, stillage_with_Parts_Weight As Double)

'Declaration
        Dim NumberStillagesPerVehicle_Volume As Integer
        Dim NumberStillagesPerVehicle_Weight As Integer

With Application.WorksheetFunction
        NumberStillagesPerVehicle_Volume =
        getNumberStillagesPerVehicle_Volume(typeTruck,
        lengthStillage, widthStillage, heightStillage, offloadingDevice,
        offloadingManner, stackability)
        NumberStillagesPerVehicle_Weight =
        getNumberStillagesPerVehicle_Weight(maxWeightVehicle,
        maxWeightTrailer, stillage_with_Parts_Weight)

        Limitation = getLimitation(NumberStillagesPerVehicle_Volume,
        NumberStillagesPerVehicle_Weight)

        getNumberStillagesPerVehicle =
        .Min(NumberStillagesPerVehicle_Volume,
        NumberStillagesPerVehicle_Weight)
End With
End Function
```

University of Pretoria etd – Van Dyk, P J S (2005)

```
Function getNumberStillagesPerVehicle_Weight(maxWeightVehicle As
Double, maxWeightTrailer As Double, stillage_with_Parts_Weight As
Double)

With Application.WorksheetFunction
        getNumberStillagesPerVehicle_Weight =
        .RoundDown(maxWeightVehicle / stillage_with_Parts_Weight, 0)
        +.RoundDown(maxWeightTrailer / stillage_with_Parts_Weight, 0)
End With
End Function

Function getLimitation(NumberStillagesPerVehicle_Volume As Integer,
NumberStillagesPerVehicle_Weight As Integer)

If NumberStillagesPerVehicle_Volume <
        NumberStillagesPerVehicle_Weight Then
        getLimitation = "Volume"
ElseIf NumberStillagesPerVehicle_Volume =
        NumberStillagesPerVehicle_Weight Then
         getLimitation = "Vol.& Weight"
Else
        getLimitation = "Weight"
End If
End Function

Function getNumberStillagesPerVehicle_Volume(typeTruck As String,
lengthStillage As Double, widthStillage As Double, heightStillage As
Double, offloadingDevice As String, offloadingManner As String,
stackability As Boolean)


'Declaration
```

```
Dim lenghtVehicle As Double
Dim widthVehicle As Double
Dim heightVehicle As Double
Dim lengthTrailer As Double
Dim widthTrailer As Double
Dim heightTrailer As Double
Dim boundaryDistanceRight As Double
Dim boundaryDistanceLeft As Double
Dim boundaryDistanceFront As Double
Dim boundaryDistanceBack As Double
Dim boundaryDistanceTop As Double
Dim NumberStillages_in_lenght_vehicle As Integer
Dim NumberStillages_in_lenght_trailer As Integer
Dim NumberStillages_in_lenght As Integer
Dim NumberStillages_in_width_vehicle As Integer
Dim NumberStillages_in_width_trailer As Integer
Dim NumberStillages_in_width As Integer
Dim NumberStillagesPerVehicle_Volume As Integer

'Input
  'LKW-Geometry:
      lenghtVehicle = getLKWGeometrie("lenghtVehicle", typeTruck)
      widthVehicle = getLKWGeometrie("widthVehicle", typeTruck)
      heightVehicle = getLKWGeometrie("heightVehicle", typeTruck)

      lengthTrailer = getLKWGeometrie("lenghtTrailer", typeTruck)
      widthTrailer = getLKWGeometrie("widthTrailer", typeTruck)
      heightTrailer = getLKWGeometrie("heightTrailer", typeTruck)

  'Boundary Condition Distances:
      boundaryDistanceRight = getBoundary("Right", offloadingDevice,
      offloadingManner)
```

```
University of Pretoria etd – Van Dyk, P J S  (2005)
        boundaryDistanceLeft = getBoundary("Left", offloadingDevice,
        offloadingManner)
        boundaryDistanceFront = getBoundary("Front", offloadingDevice,
        offloadingManner)
        boundaryDistanceBack = getBoundary("Back", offloadingDevice,
        offloadingManner)
        boundaryDistanceTop = getBoundary("Top", offloadingDevice,
        offloadingManner)

    'Calculations:
      If Sheets("Main").Range("O18").Value = False Then
      'LKW-Geometrie mit Bordwandabstaende:
        lenghtVehicle = lenghtVehicle - boundaryDistanceFront –
        boundaryDistanceBack
        widthVehicle = widthVehicle - boundaryDistanceRight –
        boundaryDistanceLeft
        heightVehicle = heightVehicle - boundaryDistanceTop

        lengthTrailer = lengthTrailer - boundaryDistanceFront –
        boundaryDistanceBack
         widthTrailer = widthTrailer - boundaryDistanceRight –
        boundaryDistanceLeft
         heightTrailer = heightTrailer - boundaryDistanceTop
      End If


    'NumberStillages_in_lenght:
    With Application.WorksheetFunction

        NumberStillages_in_lenght_vehicle = .RoundDown(lenghtVehicle
        /lengthStillage, 0) * .RoundDown(widthVehicle / widthStillage, 0)
+.RoundDown((lenghtVehicle - (.RoundDown(lenghtVehicle
```

```
University of Pretoria etd – Van Dyk, P J S  (2005)
        lengthStillage, 0) * lengthStillage)) / widthStillage, 0) *
        .RoundDown(widthVehicle / lengthStillage, 0)

        NumberStillages_in_lenght_trailer = .RoundDown(lengthTrailer /
        lengthStillage, 0) * .RoundDown(widthTrailer / widthStillage, 0) +
        .RoundDown((lengthTrailer - (.RoundDown(lengthTrailer /
        lengthStillage, 0) * lengthStillage)) / widthStillage, 0) *
        .RoundDown(widthTrailer / lengthStillage, 0)

        NumberStillages_in_lenght = NumberStillages_in_lenght_vehicle
        + NumberStillages_in_lenght_trailer

    If stackability Then
        NumberStillages_in_lenght_vehicle =
        NumberStillages_in_lenght_vehicle * .RoundDown(heightVehicle
        / heightStillage, 0)
        NumberStillages_in_lenght_trailer =
        NumberStillages_in_lenght_trailer * .RoundDown(heightTrailer /
        heightStillage, 0)
        NumberStillages_in_lenght = NumberStillages_in_lenght_vehicle
        + NumberStillages_in_lenght_trailer
    End If

'NumberStillages_in_width:
        NumberStillages_in_width_vehicle = .RoundDown(lenghtVehicle
        / widthStillage, 0) * .RoundDown(widthVehicle / lengthStillage, 0)
        + .RoundDown((widthVehicle - (.RoundDown(widthVehicle /
        lengthStillage, 0) * lengthStillage)) / widthStillage, 0) *
        .RoundDown(lenghtVehicle / lengthStillage, 0)

        NumberStillages_in_width_trailer = .RoundDown(lengthTrailer /
        widthStillage, 0) * .RoundDown(widthTrailer / lengthStillage, 0) +
```

```
        University of Pretoria etd – Van Dyk, P J S  (2005)
        .RoundDown((widthTrailer - (.RoundDown(widthTrailer /
        lengthStillage, 0) * lengthStillage)) / widthStillage, 0) *
        .RoundDown(lengthTrailer / lengthStillage, 0)

        NumberStillages_in_width = NumberStillages_in_width_vehicle +
        NumberStillages_in_width_trailer

      If stackability Then
        NumberStillages_in_width_vehicle =
        NumberStillages_in_width_vehicle * .RoundDown(heightVehicle /
        heightStillage, 0)
        NumberStillages_in_width_trailer =
        NumberStillages_in_width_trailer * .RoundDown(heightTrailer /
        heightStillage, 0)
        NumberStillages_in_width = NumberStillages_in_width_vehicle +
        NumberStillages_in_width_trailer
      End If

    'Results
        getNumberStillagesPerVehicle_Volume =
        .Max(NumberStillages_in_lenght, NumberStillages_in_width)
    End With
End Function


Function getLKWGeometrie(typeGeometry As String, typeTruck As
String)

'Declaration
With Sheets("BoundaryConditions")
Select Case typeTruck
    Case "Truck1"
```

```
    Select Case typeGeometry
        Case "lenghtVehicle": getLKWGeometrie = .Range("E8").Value
        Case "widthVehicle": getLKWGeometrie = .Range("E9").Value
        Case "heightVehicle": getLKWGeometrie = .Range("E10").Value

        Case "lenghtTrailer": getLKWGeometrie = .Range("E12").Value
        Case "widthTrailer": getLKWGeometrie = .Range("E13").Value
        Case "heightTrailer": getLKWGeometrie = .Range("E14").Value
    End Select

Case "Truck2"
    Select Case typeGeometry
        Case "lenghtVehicle": getLKWGeometrie = .Range("F8").Value
        Case "widthVehicle": getLKWGeometrie = .Range("F9").Value
        Case "heightVehicle": getLKWGeometrie = .Range("F10").Value

        Case "lenghtTrailer": getLKWGeometrie = .Range("F12").Value
        Case "widthTrailer": getLKWGeometrie = .Range("F13").Value
        Case "heightTrailer": getLKWGeometrie = .Range("F14").Value
    End Select

Case "Truck3"
    Select Case typeGeometry
        Case "lenghtVehicle": getLKWGeometrie = .Range("G8").Value
        Case "widthVehicle": getLKWGeometrie = .Range("G9").Value
        Case "heightVehicle": getLKWGeometrie = .Range("G10").Value

        Case "lenghtTrailer": getLKWGeometrie = .Range("G12").Value
        Case "widthTrailer": getLKWGeometrie = .Range("G13").Value
        Case "heightTrailer": getLKWGeometrie = .Range("G14").Value
    End Select
Case "Truck4"
```

```
        Select Case typeGeometry
            Case "lenghtVehicle": getLKWGeometrie = .Range("H8").Value
            Case "widthVehicle": getLKWGeometrie = .Range("H9").Value
            Case "heightVehicle": getLKWGeometrie = .Range("H10").Value

            Case "lenghtTrailer": getLKWGeometrie = .Range("H12").Value
            Case "widthTrailer": getLKWGeometrie = .Range("H13").Value
            Case "heightTrailer": getLKWGeometrie = .Range("H14").Value
        End Select
    End Select
End With
End Function


Function getBoundary(typeBoundary As String, offloadingDevice As
String, offloadingManner As String)

With Sheets("BoundaryConditions")
Select Case offloadingDevice
    Case "Forklift"
        Select Case offloadingManner
            Case "SideOffloading"
                Select Case typeBoundary
                    Case "Right": getBoundary = .Range("B27").Value
                    Case "Left": getBoundary = .Range("B28").Value
                    Case "Front": getBoundary = .Range("B30").Value
                    Case "Back": getBoundary = .Range("B31").Value
                    Case "Top": getBoundary = .Range("B33").Value
                End Select
            Case "BackOffloading"
                Select Case typeBoundary
                    Case "Right": getBoundary = .Range("E27").Value
```

```
                Case "Left": getBoundary = .Range("E28").Value
                Case "Front": getBoundary = .Range("E30").Value
                Case "Back": getBoundary = .Range("E31").Value
                Case "Top": getBoundary = .Range("E33").Value
            End Select
        End Select
    Case "Ameise"
        Select Case offloadingManner
            Case "SideOffloading"
                Select Case typeBoundary
                    Case "Right": getBoundary = .Range("G27").Value
                    Case "Left": getBoundary = .Range("G28").Value
                    Case "Front": getBoundary = .Range("G30").Value
                    Case "Back": getBoundary = .Range("G31").Value
                    Case "Top": getBoundary = .Range("G33").Value
                End Select
            Case "BackOffloading"
                Select Case typeBoundary
                    Case "Right": getBoundary = .Range("J27").Value
                    Case "Left": getBoundary = .Range("J28").Value
                    Case "Front": getBoundary = .Range("J30").Value
                    Case "Back": getBoundary = .Range("J31").Value
                    Case "Top": getBoundary = .Range("J33").Value
                End Select
            End Select
        End Select
    End With
End Function

Function getDeliveries_Required_per_Day(Stillages_per_day As
Integer, NumberStillagesPerVehicle As Integer)
```

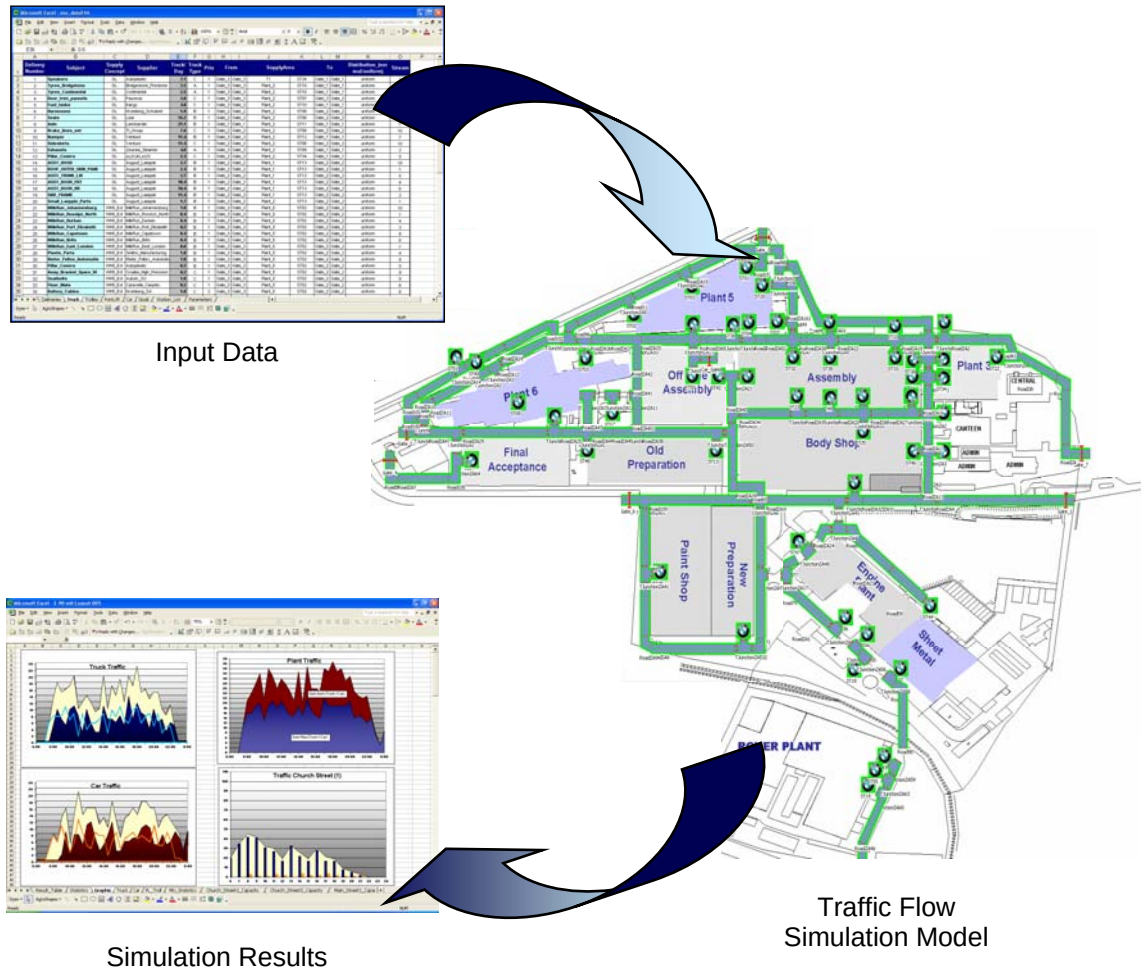
University of Pretoria etd – Van Dyk, P J S (2005)

```
    getDeliveries_Required_per_Day = Stillages_per_day /
NumberStillagesPerVehicle

End Function

Function
getVehicles_Required_per_Day(Absolute_Deliveries_required_per_day
As Integer, Delivery_Cycle_Time As Double)

'Declaration
Dim Possible_deliveries_per_day_with_1_truck As Integer

    With Application.WorksheetFunction

        Possible_deliveries_per_day_with_1_truck = .RoundUp(960 /
        Delivery_Cycle_Time, 0)
        getVehicles_Required_per_Day =
        .RoundUp(Absolute_Deliveries_required_per_day /
        Possible_deliveries_per_day_with_1_truck, 0)

    End With
End Function
```

Appendix D

Simulation Model Application

- Modeling the "New Gate 5" Scenario -



1 INTRODUCTION

To illustrate the capability and use of the tools developed during this project, one of the various changes considered for BMW Plant 9.2 in preparation of the production of the E90 (new 3 series) was evaluated and compared to the base scenario (see *9.9 Modeling Different Scenarios*).

2 SCENARIO

There are currently four gates at BMW Plant 9.2. through which vehicles enter and exit the plant (see *9.6.2 Routing Trucks* and *Figure A*).

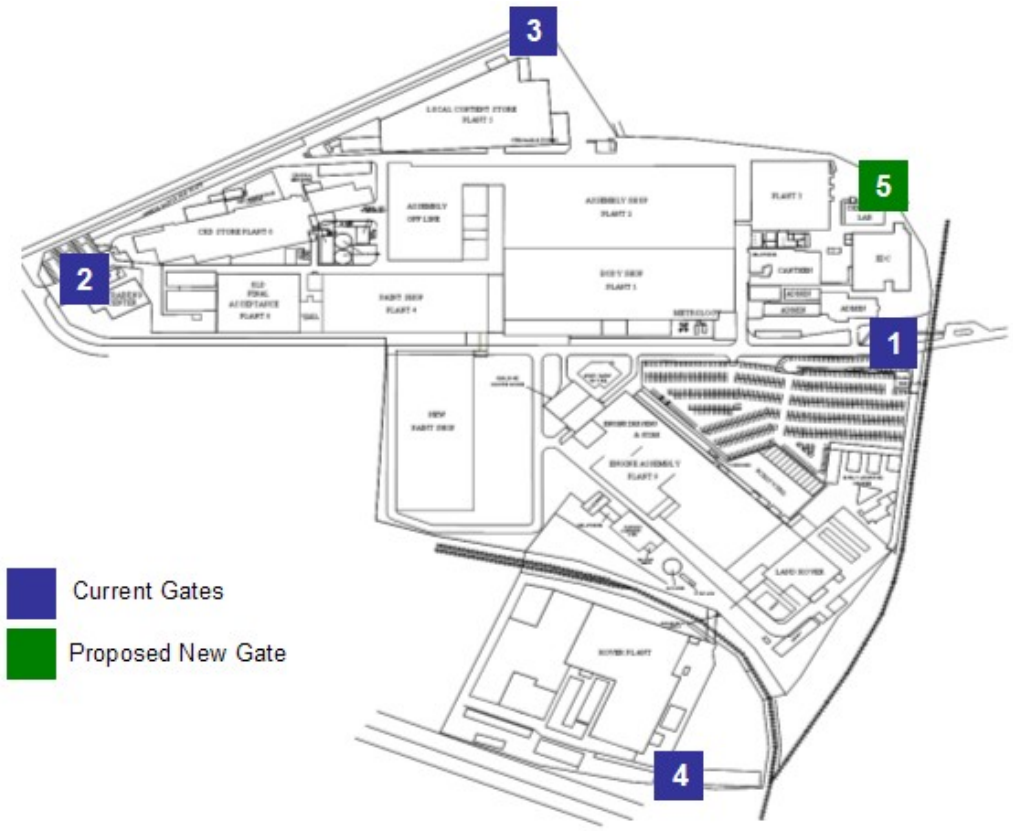


Figure A: Location of BMW Plant 9 Gates

One of the structural changes considered for BMW Plant 9.2 in preparation of the production of the E90 (new 3 series) is the introduction of a new gate, which will logically be named “Gate 5” (see Figure A). This gate will be located in the north-east side of the plant, and is expected to reduce traffic levels in both Main- and Church street, two of the (current) critical traffic flow areas within the plant (see *9.7 Simulation Output*).

The introduction of this gate will imply large expenses, like the:

- initial investment cost of constructing the new gate
- initial investment cost of building a new road stretching from Church street to the new gate, and from the new gate to Frans du Toit street outside of the plant (see Figure B)
- running cost of additional security personnel to man the new gate

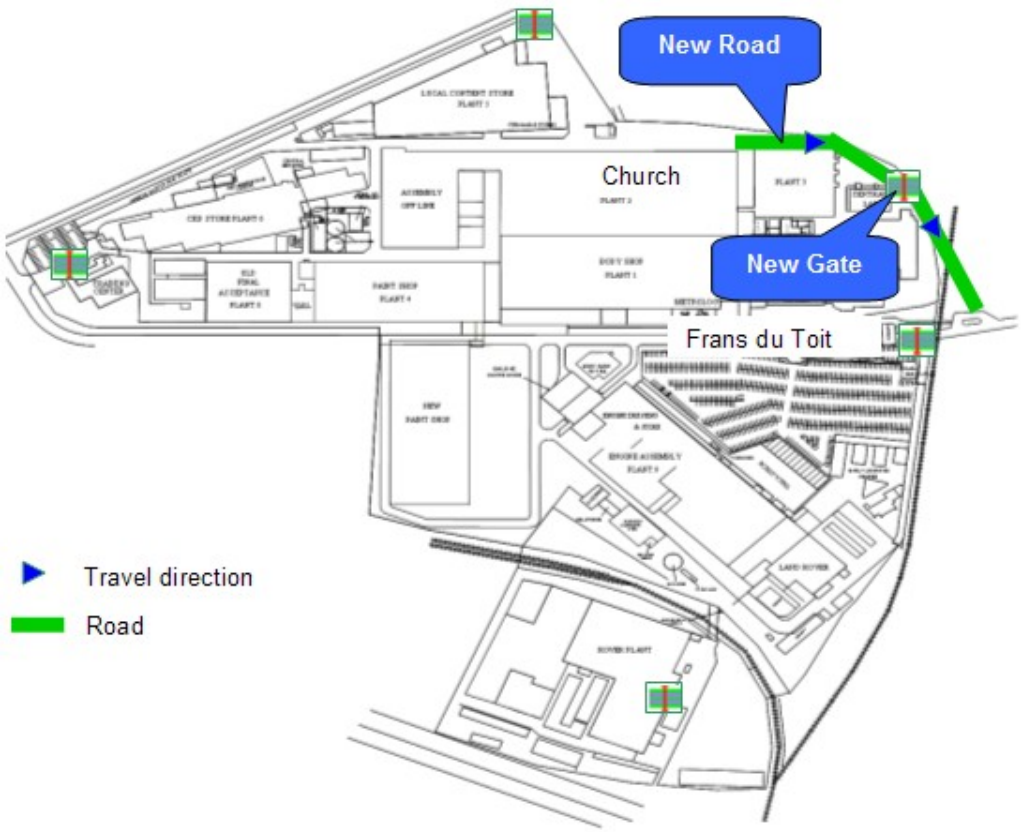


Figure B: New road- and gate construction

The new gate and –road are planned to be used as a means for trucks to exit the plant only (no vehicles will enter the plant through the new gate), creating a general flow of traffic in Church and Main street towards the new gate (see Figure C).

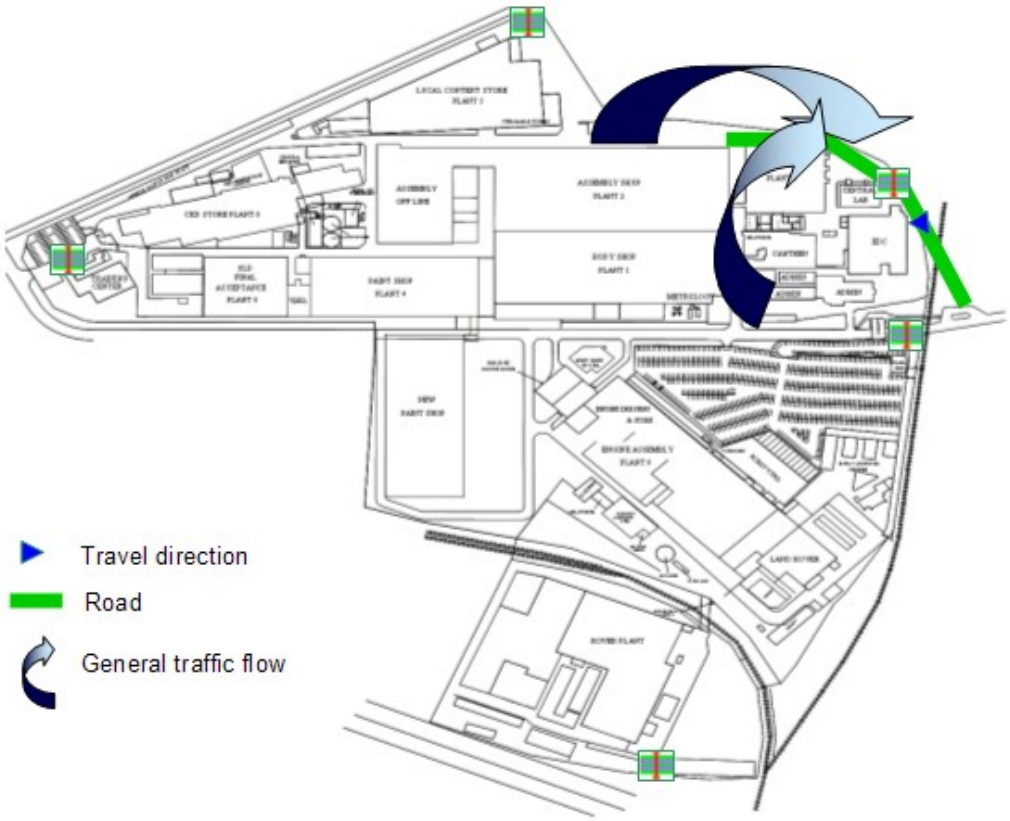


Figure C: General traffic flow towards new gate

The impact that the new gate is expected to have on the overall plant traffic could previously only be speculated upon. By utilising the tools developed during this project, the expected impact can be evaluated and quantitatively compared to the current scenario.

3 ADAPTING THE MODEL

As the input data file (see *9.3.2 Excel: Input Data*) of the current scenario will be used for this scenario as well, the SMDST will not be used. Changes to the input data may directly reflect in the simulation results. As we want to evaluate the impact of the additional road and gate on plant traffic alone, no *other* changes will be made to the model.

It is only necessary to make the following two changes to the current scenario: Firstly, the additional pre-developed gate- and roads building blocks (see *9.4.4 Model Building Blocks*) should be added to the current model's layout and connected appropriately, almost like building a puzzle. As the model is initialised, changes made to the layout will automatically be identified and updated in the rest of the model. Secondly, the *route* information of all vehicles that are to use gate 5 (the new gate) as an exit gate should be updated accordingly in the input data file (see *9.3.2 Excel: Input Data*). Every time the model is run it will automatically use the latest input data file and create unique results files for the scenario under analysis.

The traffic flow simulation model is now ready to be run. In future, *this* scenario will be referred to as the "**New Gate 5"** scenario".

4 COMPARING RESULTS

After running the *New Gate 5* scenario, it is simulation results are compared to that of the base scenario (see Figure D and Table A):

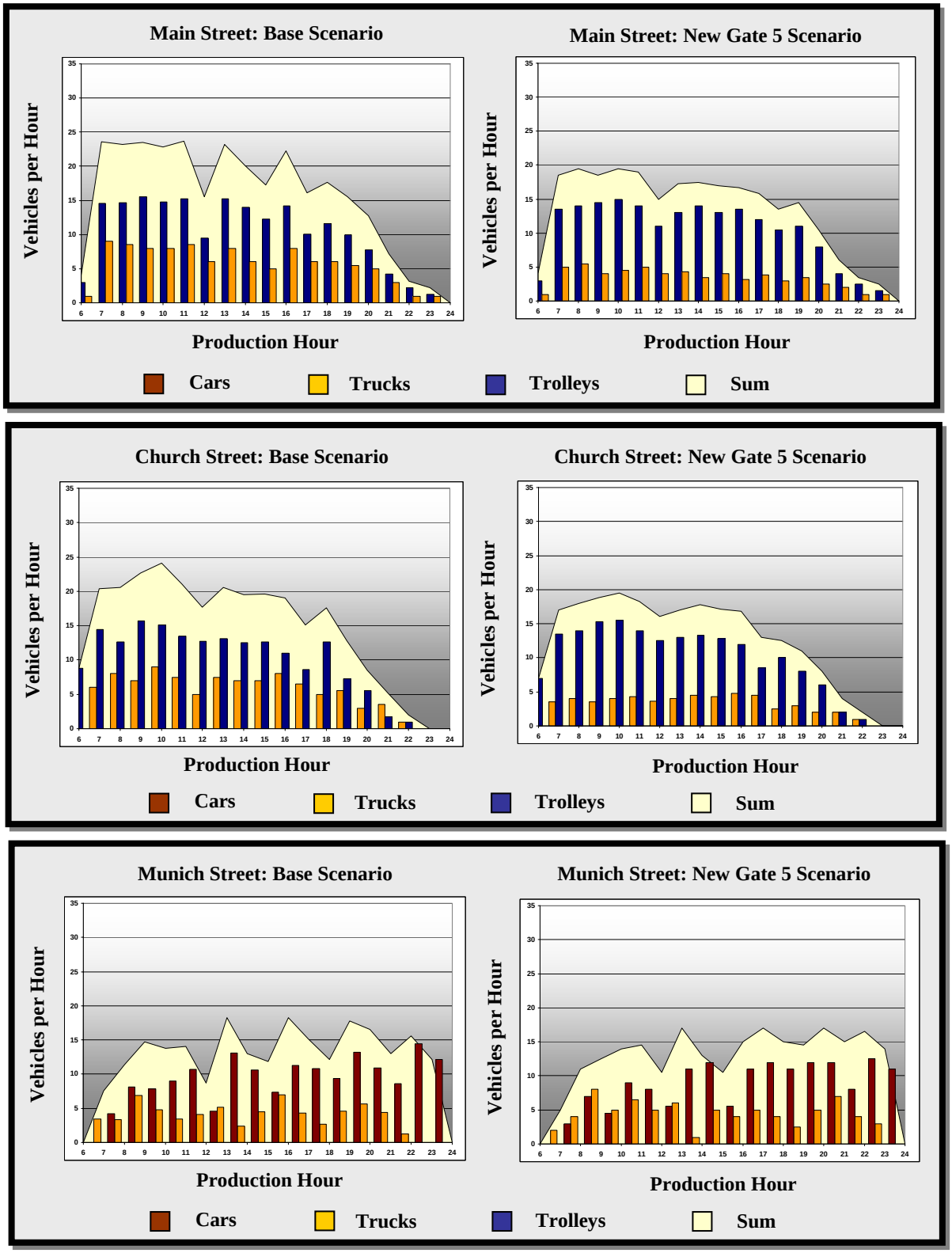


Figure D: Comparing simulation results

Table A: Comparing Base- and New Gate 5 Scenarios

General Statistics	*Scenario*	
	Base	New Gate 5
# Trucks	246.90	246.75
# Trolley (Rotations)	240.20	238.40
# Cars (left the Plant)	215.50	217.30

Car Statistics	*Scenario*	
	Base	New Gate 5
Max Process Time	06:09:40	06:13:00
Ave Process Time	02:06:50	02:06:10
Max Driven Km	8.06	7.44
Ave Driven Km	1.94	1.93

Trolley Statistics	*Scenario*	
	Base	New Gate 5
Max Supply Route (All) Km	16.39	16.47
Ave Supply Route (All) Km	7.65	7.68
Max Supply Route (Single)	2.22	2.26
Ave Supply Route (Single)	1.21	1.23

Truck Statistics	*Scenario*	
	Base	New Gate 5
Max Process Time	01:57:14	01:55:34
Ave Process Time	00:30:13	00:29:51
Max Driven Km	1.22	1.24
Ave Driven Km	0.74	0.75

5 SCENARIO CONCLUSION

The results clearly show an expected reduction in the traffic levels in Main- and Church street, but no substantial change to the traffic levels in Munich street.

Intuitively, these are exactly the results that would be expected for the New Gate 5 scenario, as:

- A considerable portion of the traffic would be directed towards the new street and –gate, reducing some of the traffic in the nearby streets
- Traffic-flow in Munich street will not be affected by the changes proposed for the New Gate 5 scenario, as all of the vehicles moving in Munich street enter and exit the plant through gate 2

As the cost of constructing the new gate and –road has already been determined, an ***informed decision*** can now be made by BMW Plant 9's management on whether this proposed scenario should be implemented or not.

This example clearly demonstrated that by utilising this DSS, scenarios can be evaluated and compared faster, more efficiently and by means of more quantitative measures than before, considerably reducing uncertainty and risk of planning

Certainly, this system will not only give BMW SA a competitive edge in preparing for the launch of the E90, but can also support other automotive manufacturers in their quest towards manufacturing excellence in an ever-increasing internationally competitive and complex environment.