Dit is een tamelijk complex object, waarin wordt gedefinieerd hoe de verschillende MessageType’s

gekoppeld worden aan de verschillende TransactionType’s. Het is mogelijk één MessageType te

koppelen aan meerdere TransactionType’s en ook omgekeerd: dezelfde MessageType meer dan eens

koppelen aan een TransactionType.

Als communicatie over meerdere transacties verloopt (en deze communicatie ook achteraf

traceerbaar dient te blijven), dan dient bij het opstellen van het raamwerk rekening gehouden te

worden met de mogelijkheid dat bij de overgang van de één naar de andere transactie meerdere

instanties van de tweede transactie gestart kunnen worden. Maar in het geval dat de rol Ober een

bestelling plaatst bij meerdere personen in de rol van Kok met T2 willen we echter niet, als we reeds

antwoord hebben gehad van één van de personen in de rol Kok, hoeven wachten totdat alle

transacties T2 zijn afgerond die op basis van het bericht msgVraag uit T1 waren geïnitieerd. We willen

zelfs met de primaire transactie T1 kunnen doorgaan zonder dat een secundaire transactie is

afgerond. In dit geval dient het element OpenSecundaryTransactionsAllowed geen waarde te

hebben. Door msgAntwoord op te nemen als BerichtInTransactie6 met zowel BerichtInTransactie5 als

BerichtInTransactie11 als previous kan de primaire transactie gewoon doorgaan.

In het geval dat de Ober op basis van msgPlaatsingBestelling uit T1 transactie T3 opstart, willen we

dat minstens één instantie van T3 is afgerond voordat de msgBestellingAkkoord wordt verzonden.

Hiervoor geven we in MessageInTransactionType het BerichtInTransactie9 de waarde true aan bij

OpenSecundaryTransactionsAllowed. Bij BerichtInTransactie8 (Bestelling niet akkoord) ligt het iets

ingewikkelder. Ervan uitgaande dat de bestelling bij meer dan één kok is uitgezet zou je kunnen

redeneren dat wanneer de eerste kok die reageert niet akkoord gaat, de ober kan afwachten of één

van de andere koks wel akkoord gaat. Ook in dat geval geldt OpenSecundaryTransactionsAllowed =

true.

Let ook op dat bij de eerste MessageInTransactionType T2 en T3 de voorgaande (previous) MITT uit

T1 komt. Het veld previous geeft aan welke MITT vooraf dient te gaan aan een andere MITT. Dit veld

is niet gekoppeld aan een specifieke transactie.

Let in het schema verder op de situatie die zich voordoet bij BerichtInTransactie6. Op basis van de

richtlijn Transactiekoppeling die in versie 1.2 van de systematiek is doorgevoerd (zie ook de

toelichting in paragraaf 1.2.1 van bijlage 5 bij de Leidraad), kan BerichtInTransactie6 worden

verzonden ondanks dat BerichtInTransactie10 ook is verzonden. De Ober kan transactie T1 dus

voortzetten zonder transactie T2 af te ronden.